Foundation Models in Autonomous Driving:
A Survey on Scenario Generation and Scenario Analysis

Yuan Gao [iD][1], Mattia Piccinini [iD][1], Yuchen Zhang [iD][1], Dingrui Wang [iD][1], Korbinian Moller [iD][1],

Roberto Brusnicki [iD][1], Baha Zarrouki [iD][1], Alessio Gambi [iD][2], Jan Frederik Totz [iD][3], Kai Storms [iD][4],

Steven Peters [iD][4], Andrea Stocco [iD][5], Bassam Alrifaee [iD][6], Marco Pavone [iD][7], Johannes Betz [iD][1]

Abstract—For autonomous vehicles, safe navigation in complex environments depends on handling a broad range of diverse and rare driving scenarios. Simulation- and scenario-based testing have emerged as key approaches to development and validation of autonomous driving systems. Traditional scenario generation relies on rule-based systems, knowledge-driven models, and data-driven synthesis, often producing limited diversity and unrealistic safety-critical cases. With the emergence of foundation models, which represent a new generation of pre-trained, general-purpose AI models, developers can process heterogeneous inputs (e.g., natural language, sensor data, HD maps, and control actions), enabling the synthesis and interpretation of complex driving scenarios. In this paper, we conduct a survey about the application of foundation models for scenario generation and scenario analysis in autonomous driving (as of May 2025). Our survey presents a unified taxonomy that includes large language models, vision-language models, multimodal large language models, diffusion models, and world models for the generation and analysis of autonomous driving scenarios. In addition, we review the methodologies, open-source datasets, simulation platforms, and benchmark challenges, and we examine the evaluation metrics tailored explicitly to scenario generation and analysis. Finally, the survey concludes by highlighting the open challenges and research questions, and outlining promising future research directions. All reviewed papers are listed in a continuously maintained repository, which contains supplementary materials and is available at GitHub.com/TUM-AVS/FM-for-Scenario-Generation-Analysis.

Index Terms—Autonomous vehicles, scenario generation, scenario analysis, foundation model, large language models.

I. INTRODUCTION

Autonomous Driving (AD) has seen rapid advancements in recent years, reaching a stage where human intervention is minimal or entirely unnecessary within specific Operational
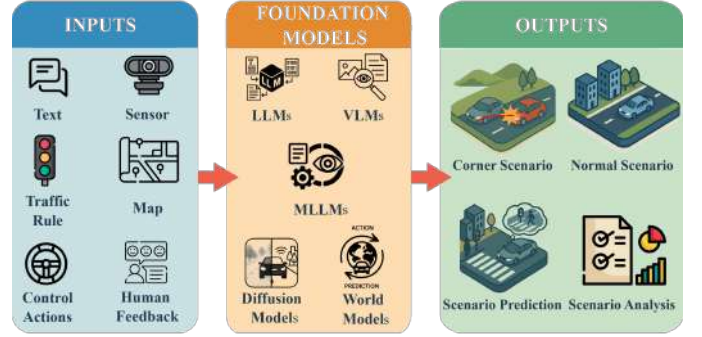

Fig. 1. This survey critically analyzes the existing Foundational Models (FMs) across Large Language Models (LLMs), Vision Language Models (VLMs), Multimodal Language Models (MLLMs), Diffusion Models (DMs), and World Models (WMs) for scenario generation and scenario analysis in autonomous driving. This fast-growing field currently exhibits numerous variants in the FMs' architectures, input types, and corresponding outputs.

Design Domains (ODDs) [1]. Companies such as Waymo have successfully deployed fully autonomous robotaxi services [2] operating at SAE Level 4 [3] since 2018, demonstrating the feasibility of driverless mobility in specific urban environments. As of 2025, Waymo serves $250,000$ commercial rides per week [4]. These advancements have been driven by the development and rigorous validation of highly reliable modular AD software functions, including perception, prediction, planning, and control [5]. In addition to the traditional modular architecture, end-to-end learning-based approaches [6], [7] have emerged, leveraging deep learning to process raw sensor data and directly generate trajectories or control actions [8].

Scenario-based testing in simulations is a key element for evaluating and validating the safety and performance of AD systems [9]. As a cost-efficient alternative to physical testing, it enables the simulation of realistic, reproducible, and controllable driving environments [10], and is particularly effective in replicating safety-critical situations, especially rare or hard-to-capture corner cases that are often absent in real-world datasets [11], [12]. Therefore, the ability to systematically generate and analyze driving scenarios is crucial to scenario-based testing, serving as a critical enabler for the development, verification, and validation of autonomous driving functions such as perception, planning, and control.

Recent advances in machine learning, especially the emergence of large-scale Foundation Models (FMs), offer new

[1] Y. Gao, M. Piccinini, Y. Zhang, D. Wang, K. Moller, R. Brusnicki, B. Zarrouki, and J. Betz are with the Professorship of Autonomous Vehicle Systems and Munich Institute of Robotics and Machine Intelligence (MIRMI), Technical University of Munich, 85748 Garching, Germany (e-mail: johannes.betz@tum.de).

[2] A. Gambi is with the Austrian Institute of Technology (AIT), Vienna, Austria (e-mail: alessio.gambi@ait.ac.at).

[3] J.F. Totz is with AUDI AG (e-mail: jan.frederik.totz@audi.de).

[4] K. Storms, S. Peters are with the Institute of Automotive Engineering (FZD) at TU Darmstadt; Department of Mechanical Engineering, 64289 Darmstadt, Germany (e-mail: steven.peters@tu-darmstadt.de).

[5] A. Stocco is with the Technical University of Munich and fortiss GmbH, Munich, Germany (e-mail: andrea.stocco@tum.de—stocco@fortiss.org).

[6] B. Alrifaee is with the Department of Aerospace Engineering, University of the Bundeswehr Munich, Germany, (e-mail: bassam.alrifaee@unibw.de).

[7] M. Pavone is with Stanford University, Stanford, CA 94305, USA (e-mail: pavone@stanford.edu).
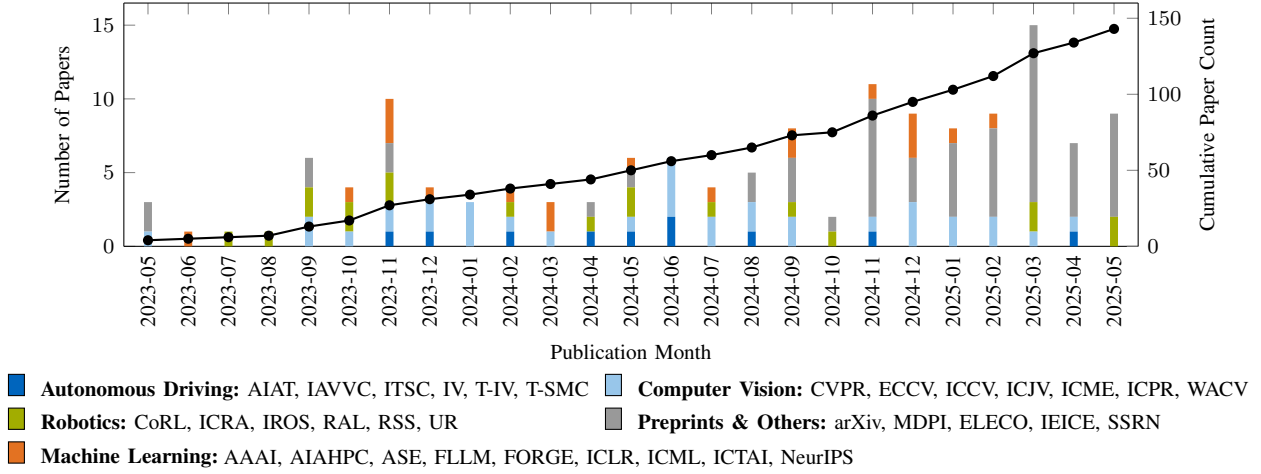
Fig. 2. Timeline of foundation model-related publications in scenario generation and analysis, across selected journals, conferences and platforms between May 2023 and May 2025. Each bar represents the monthly count of papers, grouped by thematic category. The black line shows the cumulative number of papers over time (right axis). Note that the thematic grouping refers to the general scope of the conference or journal, not to the content of the individual papers. For example, preprints listed on *arXiv* are categorized under *Preprints & Others*, although they may address topics from any of the other categories.

opportunities to enhance the realism, diversity, and scalability of scenario-based testing in autonomous driving. FMs were introduced by the Stanford Institute for Human-Centered Artificial Intelligence (HAI) in August 2021 [13] to describe a class of models trained on large-scale, diverse datasets typically using self-supervised learning. Unlike traditional machine learning models, which are often trained for specific, narrowly defined tasks, FMs can be adapted to a wide range of tasks through techniques such as prompting or fine-tuning. These models have demonstrated strong performance across various domains, including Natural Language Processing (NLP) [14], visual understanding [15], and code generation [16]. In the context of autonomous driving, FMs have recently garnered significant attention, as they combine general knowledge learned through large-scale pretraining with efficient adaptability to specific AD tasks [17]–[19].

A. Scope of the Considered Literature

In this survey, we focus on publications addressing *scenario generation* and *scenario analysis* in the context of autonomous driving with Foundation Models (see Figure 1). Our survey is based on keyword searches in Google Scholar. The full list of keywords is available in the paper's GitHub repository[1].

To ensure both breadth and relevance, we included peer-reviewed conference and journal papers as well as preprints from arXiv. Although arXiv publications are not formally peer-reviewed, they often present timely and impactful research, especially in rapidly developing areas such as FM applications. Our survey covers papers published between October 2022 and May 2025, with a primary focus on venues in autonomous driving, computer vision, machine learning/Artificial Intelligence (AI), and robotics. Figure 2 shows monthly trends in publication counts and their distribution by the thematic focus of the publication venues, e.g., conferences, journals, or preprint platforms. The complete

list of publication venues for each paper, along with the corresponding open-source code (when available), is included in the paper's GitHub repository.

B. Structure of the Survey

The structure of this survey is outlined in Figure 3. Section II provides an introduction to Foundation Models and a critical review of related surveys on scenario generation and analysis, encompassing both classical approaches and recent advances with Foundation Models. Sections III, IV, and V systematically examine language-based Foundation Models, beginning with fundamental concepts and followed by an in-depth discussion of recent applications of Large Language Models, Vision Language Models, and Multimodal Large Language Models in scenario generation and analysis. Sections VI and VII address vision-centric Foundation Models, detailing the principles of diffusion models and world models and their relevance to scenario generation. Section VIII surveys publicly available datasets and simulation benchmarks pertinent to scenario generation and analysis in autonomous driving and highlights major competition challenges in the field. Finally, Section IX and Section X identify open research questions and outline prospective research directions, while Section XI summarizes the key findings of this survey.

II. RELATED WORK & CONTRIBUTIONS

A. Development of Foundation Models

Foundation models have demonstrated versatility across a wide range of generative tasks. The term FMs, introduced in 2021 [13], refers to models trained on broad, unlabeled data at scale and adaptable to diverse downstream tasks. As illustrated in Figure 3, they support capabilities such as *Question Answering (QA)*, *image captioning*, *sentiment analysis*, *information extraction*, *object recognition*, and *instruction following*. The key element is the cross-modal adaptability of FMs which then enables generation capabilities and a deep understanding of given

[1]https://github.com/TUM-AVS/FM-for-Scenario-Generation-Analysis

Fig. 3. Overview of Foundation Models applied to scenario generation and analysis for autonomous driving, and the corresponding structure of this survey.

context. As such, foundation models and the term *generative AI* partially overlap but represent distinct concepts: the former refers to general-purpose, adaptable models; the latter to content-generating functionality.

The key moment for the wiedespread of FMs was the release of GPT-3 [20] by OpenAI in 2021, which brought **Large Language Models (LLMs)** to a broder AI community. This development was built upon the seminal work *"Attention Is All You Need"* [21], which introduced the transformer architecture and the self-attention mechanism. Unlike earlier sequential models, transformers enabled efficient parallel training over long sequences of inputs using GPUs and TPUs. Several other models built on top of Vaswani's work, including BERT [14], an encoder-only model optimized for masked language modeling; GPT [22], a decoder-only model for autoregressive generation; and T5 [23], an encoder-decoder architecture designed for text-to-text transfer. All of these models leverage self-supervised learning on massive unlabeled text corpora and serve as pre-trained backbones for diverse downstream tasks via prompting or fine-tuning.

The success of transformers quickly expanded beyond NLP. Transformer-based designs have been successfully applied to visual understanding [15], speech processing [24], tabular data [25], and multimodal learning [26]. These efforts then enabled the development of **Vision Language Models (VLMs)** such as CLIP [27] and **Multimodal Large Language Models (MLLMs)** such as LLaVA [28], which support joint text-image processing with cross-modal capabilities that resemble reasoning.

At the same time, generative modeling in vision and audio advanced significantly with the introduction of **Diffusion Models (DMs)**. The introduction of Denoising Diffusion

Probabilistic Models (DDPMs) [29] established a robust framework for generative tasks, leveraging a forward-noise process and a learned denoising reverse process to synthesize high-quality samples. DDPMs, typically trained with U-Net architectures, scale well with large, unlabeled datasets and offer strong performance across image domains. Subsequent improvements include Denoising Diffusion Implicit Models (DDIM) [30], which accelerate sampling; Latent Diffusion Models (LDMs) [31], which operate in compressed latent spaces for improved efficiency; and Video Diffusion Models (VDM) [32], which incorporate spatiotemporal structures to ensure consistency across frames.

Further, **World Models (WMs)** [33] were introduced to learn compact representations of interactive environments. These models typically consist of three components: a vision encoder (e.g., variational autoencoder (VAE)), a memory module (e.g., recurrent neural networks (RNN)), and a controller (e.g., evolution strategies). Recent advancements integrate FMs into these modules, e.g., replacing the vision or memory modules with DMs [34] or LLMs [35] to enhance their reasoning capabilities and realism. These integrated models can learn the dynamics of the environment from multimodal inputs (e.g., text, video, sensor data) and generate actionable representations for downstream tasks such as scenario generation, planning, and control.

B. Foundation Models in Autonomous Driving

Recent studies have explored the integration of FMs into AD systems, exploiting their adaptability and multimodality across both modular and end-to-end architectures. Comprehensive surveys such as [17], [18] offer a broad overview of the current landscape, covering LLMs, VLMs, MLLMs, DMs,

and WMs. While these works cover perception, planning, decision-making, simulation, and testing, they do not explicitly address the roles of FMs in scenario generation or analysis.

Regarding the overlap with generative AI, Wang et al. [36] review generative models in AD stacks. While broad in scope, their survey adopts a model-centric perspective and lacks a focused discussion on scenario generation. Diffusion models are discussed without distinction from world models, and language model applications are restricted to QA tasks. As a result, their survey lacks depth on FM-driven scenario generation and analysis, which this review aims to address.

1) LLMs in Autonomous Driving: The survey by Zhu et al. [37] reviews the integration of LLMs into modular autonomous driving systems, and focuses on perception, decision-making, control, and end-to-end approaches. Similarly, Wu et al. [38] investigate the use of LLMs for multi-agent perception, decision-making, and simulation. However, both surveys primarily focus on the broader scope of autonomous driving tasks and offer only limited coverage of scenario generation. For instance, Zhu et al. [37] briefly mention dataset generation, while Wu et al. [38] address scenario generation only in the context of vehicle–assistant interaction. Finally, Li et al. [39] review the role of LLMs in enabling human-like reasoning in both modular and end-to-end AD systems and also emphasize training and integration strategies, which is not relevant to our tasks.

2) VLMs in Autonomous Driving: The survey [40] explores the use of VLMs across a range of autonomous driving tasks, with scene understanding, visual reasoning, and dataset generation involving diffusion and world models. While it mentions several works related to scenario generation, the discussion lacks categorization and technical depth, offering no detailed analysis of the datasets, model architectures, or generation techniques involved.

3) MLLMs in Autonomous Driving: Cui et al.'s [41] survey focuses on MLLMs architectures, modality fusion, and their applications across AD tasks. Fourati et al.'s [42] survey introduces XLMs as a combination of LLMs, VLMs, and MLLMs, providing a review of their use in AD that covers concepts, workflows, and techniques. Finally, Li et al's [43] survey explores LLM and MLLM applications across different AD modules, covering integration and training techniques. However, in the three surveys [41]–[43], scenario generation is mentioned briefly and is not in the main scope.

4) DMs and WMs in Autonomous Driving: Guan et al. [44] provide an overview of world models in AD, focusing on their applications in scenario generation, motion prediction, and control. Driving WMs are further explored by Tu et al. [45], which categorize them into 2D scene evolution, 3D occupancy prediction, and scene-free paradigms. However, both surveys lack a clear classification of practical applications for scenario generation and do not differentiate between DMs and WMs, nor do they provide an in-depth technical analysis of the underlying methods.

C. Scenario Generation in Autonomous Driving

Scenario formats in autonomous driving range from annotated sensor data and multi-camera streams to map-based layouts, simulated urban scenes, and traffic-level environments, e.g., OpenScenario[2]. Examples of driving scenarios in different formats are illustrated in Figure 4.

1) Surveys with Classical Approaches: Most of the existing reviews deal with classical methods (i.e., not FM-based) for scenario generation. Nalic et al [52] introduce knowledge-driven and data-driven generation approaches, and discuss safety metrics for scenario assessment. They also propose a six-layer model, which captures all essential components of a scenario. Ding et al. [53] categorize scenario generation methods into data-driven, adversarial, and knowledge-based approaches, providing detailed insights into the mechanisms underlying each. They also highlight the role of deep generative models for synthesizing image- and video-based scenarios with several papers across different models. In alignment with the ISO/WD PAS 21448 standard[3] — Safety of the intended functionality (SOTIF), Schutt et al. [54] examine scenario generation across functional, logical, and concrete levels of abstraction. Their review includes machine learning-based generation, optimization-driven scenario exploration, scenario extraction from driving data, and manual scenario design.

2) Surveys with Foundation Models:

Survey with FMs: Huang et al. [17] provide an overview of various types of foundation models and briefly discuss scenario generation. However, their analysis is limited to input modalities and model types, without addressing specific techniques or evaluation strategies.

Surveys with VLMs: Yang et al. [55] examine the use of LLMs and VLMs in tasks such as perception, question answering, and generation. They mention scenario generation using VLMs, DMs, and WMs but provide no clear distinction between these model types. While several evaluation metrics are cited, these are not organized by task or application. Tian et al. [56] present a more structured review of VLMs in autonomous driving across LLMs, VLMs, and WMs, focusing particularly on traffic simulation via VLM-guided generation and their integration with diffusion models. However, the survey lacks information about input modalities, scenario generation techniques, and the distinction of model types.

Surveys with DMs and WMs: Fu et al. [57] review video generation and WMs, covering diffusion-based, autoregressive, and reinforcement learning approaches. Feng et al. [58] focus on WMs, categorizing outputs into images, bird's-eye views, and 3D point clouds, and discuss evaluation metrics such as semantic segmentation and occupancy prediction. However, neither survey directly connects model outputs to scenario generation tasks. They also fail to distinguish between standalone DMs and WMs, and lack discussion of concrete techniques and evaluation strategies.

---

[2]https://www.asam.net/standards/detail/openscenario/
[3]https://www.iso.org/standard/77490.html

Fig. 4. Examples of driving scenarios in autonomous driving: datasets and simulations used for scenario-based testing. Sensor data such as camera images, videos, and LiDAR point clouds derived from these scenarios can be used to evaluate perception algorithms. Concurrently, simulator-specific scenario formats support rigorous testing of planning and control modules.
Top row (left to right): Waymo Open motion [46] dataset; Argoverse2 [47] dash camera video; NuPlan [48] multi-camera views with map overlays.
Bottom row (left to right): CommonRoad [49] motion planning scenario; CARLA [50] simulated urban scenario; SUMO [51] large-scale traffic scenario.

D. Scenario Analysis in Autonomous Driving

Scenario analysis means the systematic evaluation of driving scenarios as shown in Figure 4. It encompasses tasks such as scenario evaluation, scene understanding, risk assessment, anomaly detection, and accident prediction. Further, it includes identifying safety-critical situations, evaluating system robustness, and supporting informed decision-making in both simulation and real-world environments.

1) Surveys with Classical Approaches: Riedmaier et al. [10] propose a taxonomy of scenario-based safety assessment methods, including knowledge-based, data-driven, and falsification-based approaches. They emphasize the use of key performance indicators (e.g., time-to-collision, required deceleration) as proxies for accident risk and advocate for the integration of formal methods into safety validation.

Mahmud et al. [59] review proximal surrogate indicators such as time-to-collision, post-encroachment time, and deceleration rate to avoid a crash. They categorize these metrics and identify key research challenges, including metric standardization, real-world validation, and integration into simulation-based scenario analysis frameworks.

2) Surveys with Foundation Models:

Survey with FMs: Huang et al. [17] briefly mention scenario analysis under the term "perception data annotation", but do not categorize tasks based on their goals. Additionally, they neither associate datasets with individual studies nor discuss modality transformations; as such, their review does not focus on pre-trained FMs.

Surveys with VLMs: Yang et al. [55] address scenario analysis in the context of question answering, focusing mainly on dataset descriptions. Their analysis remains limited, as it lacks discussion of input modalities, methodological approaches, model taxonomy, and performance evaluation metrics. Similarly, Tian et al. [56] consider Visual Question Answering (VQA) as a form of scenario analysis using VLMs, but cover a small number of resources and provide minimal discussion of models or methods.

E. Critical Summary

To the best of our knowledge, existing surveys on FMs in autonomous driving and scenario generation (summary in Table I) are limited in the following aspects:

- **Lack of focus on scenario generation:** None of the reviewed surveys explicitly focuses on scenario generation using FMs. When addressed, scenario generation is either mentioned briefly (e.g., [17] and [43]) or discussed without in-depth analysis of generation techniques, scenario controllability, or evaluation metrics (e.g., [55], [57], and [56]).
- **Incomplete coverage of scenario analysis:** Tasks such as scenario understanding, evaluation, and risk assessment are overlooked. When addressed (e.g., [55] and [56]), the analysis is typically reduced to question answering, with little attention paid to task-specific models, methods, or evaluation strategies.
- **Limited connections among modalities and tasks:** While several surveys consider the input modalities of FMs (e.g., text, image, and vehicle sensor data), they do not establish clear links between these modalities and the techniques, models and datasets used for scenario generation and analysis.
- **Absence of a structured classification:** No prior work presents a structured classification of FMs that spans both scenario generation and scenario analysis, considering pre-trained model types, adaptation methods (e.g., prompting, fine-tuning), input modalities, datasets, and evaluation metrics.

TABLE I
COMPARISON OF SURVEYS ON FMS FOR SCENARIO GENERATION AND ANALYSIS IN AD. OUR SURVEY IS THE FIRST TO COVER ALL FM TYPES, SCENARIO CATEGORIES, INPUT MODALITIES, DATASETS, MODEL TYPES, TECHNIQUES, AND EVALUATION METRICS.

Survey	LLM	VLM	MLLM	DM	WM	Scenario Generation[1]								Scenario Analysis[2]						
						Scenario Category	Input Modality	Dataset	Scenario Controllability	Model	Technique	Metric	[n/m]	Scenario Category	Input Modality	Dataset	Model	Technique	Metric	[n/m]
2023 Huang. [17]	✓	✓	✓	✓	✓		✓			✓			13/261		✓		✓			5/261
2024 Yang. [55]	✓	✓					✓				✓		11/155	✓			✓			13/155
2024 Fu. [57]				✓			✓			✓			11/114							
2024 Tian. [56]	✓	✓		✓		✓		✓		✓			15/124		✓		✓			9/124
2025 Feng. [58]				✓						✓	✓		13/166							
2025 **Our Work**	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	**90/332**[*]	✓	✓	✓	✓	✓	✓	**53/332**[*]

[1] **Scenario Category**: safety-critical scenario, ADAS test scenario; **Input Modality**: text, image, sensor signal (e.g., LiDAR), video; **Dataset**: e.g., nuScenes;
   Scenario Controllability: full (script), partial (trajectory); **Model**: e.g., GPT, latent diffusion; **Technique**: e.g., prompting, fine-tuning; **Metric**: e.g., realism, safety.
[2] **Scenario Category**: e.g., evaluation, risk assessment; **Input Modality**: text, image, sensor signal (e.g., LiDAR), video; **Dataset**: e.g., nuScenes; **Model**: e.g., GPT, LLaVA;
   Technique: e.g., zero-shot, adapter layer; **Metric**: e.g., accuracy, language generation quality.
   [n/m] = number of papers using FMs (large pretrained models) / total papers reviewed in the survey.
[*] The 332 papers are categorized as follows: 90 on scenario generation, 53 on scenario analysis, 55 on datasets, 21 on simulators, 25 on benchmark challenges, and 88 on other related topics (e.g., FMs' implementation).

F. Contributions

To address the limitations of the existing literature reviews, this survey offers a full evaluation on the landscape of FMs in the field of scenario generation and scenario analysis (see Table I). In summary, this work provides the following key contributions:

1) **Review of scenario generation and analysis using FMs**: We present the first survey on scenario generation and analysis in autonomous driving covering 143 papers using FMs, covering LLMs, VLMs, MLLMs, DMs, and WMs.

2) **Structured classification of existing methods**: Our work offers a structured classification covering all FM types (i.e., LLMs, VLMs, MLLMs, DMs, WMs), scenario categories, input modalities, model types, datasets, techniques, and evaluation metrics.

3) **Review of datasets, simulation platforms and existing benchmarking competitions**: We review the openly-accessible datasets and simulators used for scenario generation and analysis. Meanwhile, we provide the first review on benchmarking competitions for FMs in autonomous driving.

4) **Identification of open challenges and future directions**: We identify key open research challenges in applying FMs to scenario-based testing. By leveraging our analysis, we propose future research directions to improve the adaptability, robustness, and evaluation of FM-driven approaches in scenario generation and scenario analysis.

III. LARGE LANGUAGE MODELS (LLMS)

This section introduces the foundation and evolution of LLMs, presents key technological advancements, and reports on common adaptation techniques (e.g., prompt engineering and fine-tuning strategies). We then explore how LLMs support scenario generation, safety-critical cases, real-world scene synthesis, driving policy evaluation, closed-loop simulation, and Advanced Driver Assistance Systems (ADAS) testing. The section concludes with a discussion on scenario analysis, including question answering, scenario understanding, and scenario evaluation.

A. Development of LLMs

1) Evolution of LLMs: The most prominent category of FMs are LLMs, which focus on the text modality and are built upon the transformer architecture [21]. The transformer consists of an encoder-decoder structure featuring Multi-Head Attention and Feedforward Neural Networks (FFN). Multi-head attention enables the model to capture diverse contextual dependencies by attending to different parts of the input simultaneously, while the FFN projects these representations into a shared feature space.

A key advancement toward self-supervised learning came with the introduction of static word embeddings [60], laying the groundwork for pre-trained language models. This led to context-aware models like GPT [22], BERT [14], and T5 [23], which replaced static embeddings with dynamic representations and learned directly from unlabeled text. The release of GPT-3 marked a significant turning point. With 175B parameters and training on a massive corpus [20], it demonstrated generalization and few-shot learning abilities, significantly reducing the need for task-specific fine-tuning compared to earlier models like GPT-1 (117M parameters) and GPT-2 (1.5B parameters).

A key driver of LLMs progress has been the rise of large-scale AI accelerators (e.g., NVIDIA A100, Google TPU), enabling efficient training of massive models. OpenAI's 2020 scaling laws [61] showed that LLMs performance improves predictably with increased model size, data, and compute, fueling the trend toward ever-larger foundation models. However, DeepSeek [62] challenged this assumption by demonstrating that data quality and alignment matter as much as scale. Through supervised fine-tuning on synthetic expert data and reinforcement learning via Group Relative Policy Optimization (GRPO), they trained smaller models that achieved competitive performance.

Recently, LLMs have been integrated throughout the ADs pipeline [37]–[39], encompassing both modular and end-to-end architectures. Their reasoning and inference capabilities, along with their adaptability to task-specific objectives through the text modality, contribute to enhanced robustness, interpretability, and adaptability in autonomous systems.

2) Adaptation Techniques for LLMs: Since LLMs are pre-trained on large-scale unlabeled data, various adaptation techniques such as prompt engineering and fine-tuning have been developed to tailor LLM's behavior to specific tasks. Sahoo et al. [63] provide an overview of these techniques in their recent survey. Here, we focus on the adaptation techniques that were applied in the context of driving scenario generation and analysis.

Prompt Engineering: Prompt Engineering refers to designing and structuring input prompts to guide a pre-trained language model toward producing desired outputs, without modifying its internal parameters. The different techniques are:

(1) Chain-of-Thought (CoT): Encourages the model to generate intermediate reasoning steps before producing a final answer. This structured reasoning enhances the logical consistency of the model's reasoning chain, which is particularly beneficial for complex, multi-step tasks.

(2) In-Context Learning (ICL): Involves task demonstrations (e.g., *zero-shot*, *one-shot*, or *few-shot*) in the prompt to guide the model towards the correct task behavior.

(3) Self-Consistency (SC): A decoding strategy that samples multiple outputs for a given prompt and selects the most frequent or consistent one, improving answer robustness and reliability.

(4) Retrieval-Augmented Generation (RAG): Enhances the performance on specific tasks by retrieving external knowledge from a database at inference time. A retrieval component identifies the relevant documents to condition the model's response, thereby improving its accuracy.

(5) Contextual Prompting (CP): Augments prompt with task-specific context or background information, thereby helping the model align more closely with the intended application domain.

Fine-Tuning: These techniques train the model on datasets of instruction–response pairs to improve its ability to follow natural language instructions. The different fine-tuning techniques are:

(1) Full Fine-Tuning: Updates all model parameters using domain-specific data. While effective, it requires significant computational resources and has limited scalability.

(2) Parameter-Efficient Fine-Tuning (PEFT): Updates only a small portion of the model's parameters, while keeping most of the model frozen. A specific PEFT method is *LoRA (Low-Rank Adaptation)*, which injects trainable low-rank matrices into the attention modules of the model, enabling adaptation with minimal parameter updates and reducing computational cost. For instance, full fine-tuning of GPT-3 requires updating all 175B parameters, whereas LoRA can achieve comparable performance by training only around 37.7M parameters [94].

Additionally, more advanced techniques to adapt LLMs exist in the reviewed papers. These include *multi-stage prompting* [86], [90] and *multi-LLM agent systems (MLAs)* [68], which coordinate multiple interacting LLMs to solve complex tasks collaboratively. Tooling frameworks such as LangChain [95] facilitate the construction of modular, agent-based architectures that extend beyond traditional single-prompt interactions.

B. LLM-Based Scenario Generation

Advances in LLMs have triggered the development of LLM-driven scenario generation to test intelligent vehicle systems. Based on their individual objectives, we classify the existing works into six categories and list representative works within each category in Table II:

1) Safety-Critical Scenario Generation: A key application of LLM-based scenario generation lies in the creation of safety-critical scenarios. Often termed "corner cases", "long-tail", or "out-of-distribution (OOD)" events, these scenarios are crucial for evaluating the safety and robustness of AD systems [11].

The work LLMScenario [66] focuses on safety-critical scenario generation based on the HighD dataset [64] with GPT-4. They use ICL, incorporating bad examples, which are evaluated based on reality and rarity, to guide the generation. Using CoT and SC prompting, the framework of [65] generates safety-critical trajectories step-by-step in MetaScenario. ChatScene [67] uses GPT-4 with RAG to translate textual safety-critical descriptions into the domain-specific language (DSL) like Scenic [96] for CARLA [50], focusing on complex urban environments. ChatScene's retrieval database is constructed using Sentence T5 embeddings, which map sub-components (e.g., behaviors and geometry) to code snippets. Then these code snippets are assembled into Scenic scripts, which could be simulated in CARLA. Building on structured generation, Aasi et al. [68] propose a multi-agent pipeline that constructs a branching tree of OOD scenarios using CoT prompting. Their Augmenter-LLM, based on GPT-4o, translates descriptions into CARLA scene configurations, which contain maps, weather, objects, and behaviors via API calls. A VLM then classifies the simulated scenes by OOD type.

The methods discussed above operate open loop. In contrast, Mei et al. [69] focus on online interactive scenario generation using Waymo Open Motion Dataset [46]. Their retrieval-augmented framework uses DeepSeek-V3 and DeepSeek-R1 to infer risky behaviors of a vehicle in real time and synthesize adversarial trajectories for it to collide with the ego vehicle. A memory module stores and retrieves intent–planner pairs, allowing continuous refinement and adaptation of the generated scenarios.

Despite promising advances, current works often operate offline or focus on limited risk types, limiting their generalizability to complex, multi-agent contexts. Future work could integrate interactive generation, enhance safety verification in simulation, and develop evaluation pipelines by leveraging VLMs to assess the plausibility and criticality of the generated scenarios.

Metrics: Safety-critical scenario generation is evaluated using (I) *Safety-criticality metrics*, such as time-to-collision [69] and collision rate [67], [69], quantify the likelihood of collisions involving the ego vehicle; (II) *Diversity and OOD metrics*, including rarity and reality [66], OOD-ness and diversity [68], to measure the distributional shift, scenario variety, and physical plausibility.

TABLE II
SUMMARY OF SCENARIO GENERATION STUDIES USING LARGE LANGUAGE MODELS.

Category	Input				Model	Technique[1]	Simulator	Output[2]	Paper
	Trajectory	DSL	Dataset	Database					
Safety-critical scenario (Sec III-B1)	✓		HighD [64]		GPT-4	CoT, ICL, SC	Metascenario [65]	(trajectory)	LLMScenario [66]
		✓		Map Position Behaviors	GPT-4	CoT, ICL, RAG	CARLA [50]	(script)	ChatScene [67]
			CARLA		GPT-4o Claude 3.5 Sonnet Gemini 1.5 Pro	CoT, MLAs	CARLA	(script)	Aasi et al. [68]
	✓		WOMD [46]	Trajectory Behaviors	DeepSeek V3 DeepSeek R1	CoT, CP, ICL, RAG	WOMD	(script)	Mei et al. [69]
Real-world scenario (Sec III-B2)	✓		Interaction [70]		GPT-4	CoT	Interaction	(trajectory)	SeGPT [71]
			Waymo Open [72]	Map NHTSA [73]	GPT-4	CoT, ICL, RAG	MetaDrive [74]	(trajectory)	LCTGen [75]
			HighD		GPT-4	CoT	Esmini CarMaker	(script)	Chat2Scenario [76]
			SUMO, OSM [77]	Map	Llama 3.1-8b	CoT, RAG	SUMO [51]	(script)	ChatSUMO [78]
Driving policy testing (Sec III-B3)			CARLA	Map Position Behaviors	GPT-4o	CoT, RAG	CARLA	(script)	TTSG [79]
		✓	CARLA		GPT-4	CoT, CP	CARLA	(script)	AutoSceneGe [80]
Closed-loop scenario (Sec III-B4)			Waymo Open	Map States	Llama 3.1-8b	LoRA	Waymo Sim	(trajectory)	ProSim [81]
	✓		Waymo Open MetaDrive [74]		Llama 3.1-8b	CoT, RL	MetaDrive	(trajectory)	LLM-attacker [82]
	✓		HighD		Mistral-7B	CoT, Langchain	HighwayEnv [83]	(trajectory)	CRITICAL [84]
Dataset (Sec III-B5)			Waymo Open	Images	GPT-4	MLAs		(video)	Chatsim [85]
ADAS test scenario (Sec III-B6)		✓		Traffic rule	GPT-4	Multi-stage, ICL, CoT, CP	CARLA	(script)	TARGET [86]
		✓		Regulation	GPT-4	ICL	SUMO	(script)	Guzay et al. [87]
			LGSVL [88]	NHTSA	GPT-4	CP, ICL	LGSVL	(script)	SoVAR [89]
		✓		NHTSA	GPT-4	CP, ICL, Multi-stage	LGSVL	(script)	LeGEND [90]
		✓		Standard Regulation Test Specification	GPT-4 Llama 3	CP, Multi-stage	CARLA	(script)	Petrovic et al. [91]
		✓		NHTSA OpenXOntology	GPT-4	CoT, ICL, SC, Multi-stage	CARLA	(script)	Text2Scenario [92]
		✓	OpenDRIVE	UNECE R157 OpenX Ontology maps	Llama 3.1	CP	VTD	(script)	Zhou et al. [93]

[1] Techniques: CoT = Chain-of-Thought prompting; ICL = In-Context Learning; SC = Self-Consistency; CP = Contextual Prompting; RAG = Retrieval-Augmented Generation; RL = Reinforcement Learning; LoRA = Low-Rank Adaptation; MLAs = Multi-LLM Agent Systems.

[2] Output: (video) Video, (trajectory) Trajectory, (script) Scenario script.

2) Real-World Scenario Synthesis: Creating realistic driving scenarios is challenging due to the difficulty of matching real-world conditions. A common strategy to synthesize these scenarios involves replaying recorded driving data in simulation environments or leveraging real crash reports to reconstruct the corresponding events. Realistic traffic scenes can also be synthesized by grounding them on real-world maps, which helps to preserve the authentic infrastructure, road layouts, and environmental features.

LCTGen [75] leverages GPT-4 with ICL and CoT prompting to convert crash report into structured YAML-like descriptions. Then a retriever module matches these structured descriptions with relevant maps from the Waymo Open Dataset [72]. These "map-grounded" inputs are then processed by a generative model using multi-layer perceptrons and learned masks to produce realistic driving scenarios. In Chat2Scenario [76], recorded datasets with user-defined criticality and textual descriptions are used as input. They use a templated contextual prompting scheme with GPT-4

and retrieve relevant scenarios that match the user's input with ASAM OpenScenario [97] format. Then visualize them using Esmini[4] and CarMaker[5]. For microscopic simulation, ChatSUMO [78] utilizes Llama 3.1 [98] with template-based prompts to extract user requirements for traffic volume, city, and network type. Then, ChatSUMO translates these parameters into SUMO [51] configurations, with OpenStreetMap (OSM) [77] maps retrieved through RAG. Simulation outputs, including traffic density, travel time, and emissions, are visualized and summarized via a Streamlit interface[6]. Finally, SeGPT [71] synthesizes diverse and challenging test data from recorded trajectories. They use GPT-4 to generate multi-agent trajectories. Their framework supports large-scale scenario synthesis and compares zero-shot prompting with CoT to evaluate LLM-guided generation performance on the Intersection dataset [70].

[4] https://esmini.github.io/
[5] https://www.ipg-automotive.com/
[6] https://streamlit.io/

Reviewed papers generate scenarios from recorded data and crash reports. One of the future directions is to first generate scenarios from recorded datasets and then incorporate natural language descriptions as feedback. This hybrid approach could significantly enhance both the realism and diversity of the resulting scenarios by aligning data-driven generation with human-intuitive requirements.

Metrics: Evaluation of real-world scenarios considers the following: (I) *Trajectory accuracy*, which can be quantified with the mean Average Displacement Error (mADE), mean Final Displacement Error (mFDE), and Maximum Mean Discrepancy (MMD), to measure how closely simulated behaviors match the ground truth [71], [75]; (II) *Semantic correctness*, which uses classification metrics like accuracy, F1 score to assess alignment with crash report descriptions [76]; and, (III) *Traffic realism*, which analyzes the traffic density, travel time, fuel consumption, and emissions to evaluate map-based simulations [78].

3) Driving Policy Test Scenario Generation: Driving policy test scenario is crucial for evaluating the behavior of specific AD policies, such as motion planners or controllers. In LCTGen [75], generated real-world crash scenarios are used to assess the performance of a motion planner within the MetaDrive [74] simulator. In TTSG [79], GPT-4o-generated scenarios are used for multi-agent planning validation in critical scenarios. Specifically, they constructed a road and agent database using RAG with LLMs and proposed ranking strategies to select the best-fitting road based on the agent's behavior. In contrast, AutoSceneGen [80] uses ICL to prompt GPT-4 to directly generate a DSL-style configuration code compatible with CARLA. AutoSceneGen uses a code-designed filter to select the valuable part of the scenario description based on simulation documents and ICL, and then adds scenario examples to the prompt. Then, a code-based validator transfers and validates the output from GPT-4.

Metrics: Driving policy performance is commonly evaluated through: (I) *Trajectory accuracy*, quantified with ADE and FDE [80] capturing deviations between predicted and ground-truth trajectories; (II) *Policy robustness*, using collision rate and overall scores, for instance, using the CARLA SafeBench to assess safety and resilience of policies under diverse conditions [79].

4) Closed-loop Scenario Generation: Recent works address the limitations of static datasets by introducing closed-loop scenario generation with LLMs. Closed-loop scenarios enable the validation of multi-agent interactions and ego-reactive behaviors.

ProSim [81] presents a promptable closed-loop simulation framework, where prompts such as goal points, route sketches, action tags, and natural language instructions are used to guide an agent's behavior. Llama3.1-8B is fine-tuned with LoRA to generate policy tokens, and a lightweight policy module rolls out the agent's trajectories in a closed loop within the Waymo simulator. In LLM-attacker [82], an adversarial scenario generation is proposed. It employs three coordinated modules based on Llama3.1-8B, for initialization, reflection,

and modification, to identify and refine adversarial vehicle behaviors using CoT. These modules iteratively generate and adjust the attacker's trajectories to collide with the ego vehicle. Their framework is trained with reinforcement learning in a closed-loop setting using the Waymo Open Dataset. In contrast, CRITICAL [84] focuses on ego-agent policy learning without adversarial agents. It integrates Mistral-7B via LangChain [95] into a standard reinforcement learning loop in the HighwayEnv environments [83]. Their LLM is used to generate diverse scenario configurations, such as vehicle density, number of cars, and to shape safety-related rewards, enabling robust policy learning under different conditions.

Together, these works demonstrate complementary strategies: ProSim enables fine-grained control and interactivity, LLM-Attacker focuses on adversarial testing, and CRITICAL supports LLM-guided training environments. Future research could benefit from unifying these paradigms into a single framework that supports diverse behavior modeling, adversarial robustness, and controllable training environments.

Metrics: Evaluation in closed-loop simulation is typically done with (I) *trajectory accuracy*, assessed via displacement-based metrics such as ADE and FDE [81], [82], and (II) *controllability*, defined as the relative improvement in ADE when prompts are applied, thus measuring how well an agent's behavior follows user-specified intentions [81].

5) Datasets Generation: Real-world camera datasets are widely used in autonomous driving research, but often lack the diversity and editability required for generating specific test cases. To address this, recent work explores language-guided editing of recorded images.

ChatSim [85] introduces a collaborative multi-agent framework with GPT-4, where each LLM agent handles a specialized scene editing task, such as viewpoint changes, vehicle manipulation, asset insertion, and motion planning, based on natural language instructions. ChatSim leverages neural rendering and lighting estimation to achieve photorealistic, multi-camera scene synthesis with external digital assets.

Metrics: The evaluation of generated datasets focuses on (I) *semantic alignment*, which assesses how closely the generated content matches the input command [85], and (II) *visual quality*, measured by pixel-level fidelity, structural similarity, and perceptual similarity [85]. Most metrics focus on (III) *human-perceived realism*, such as pixel-level fidelity and structural similarity, but do not necessarily reflect how well scenes support perception tasks in autonomous systems.

6) ADAS Test Scenarios Generation: To evaluate the performance of ADAS systems, it is necessary to translate abstract functional scenarios, often derived from regulations or specifications, into executable test cases. Recent works leverage LLMs to automate this process by extracting structure and semantics from text and generating scripts for simulators.

One of the pioneering papers on ADAS test scenario generation using LLMs is Guzay et al. [87], who use GPT-4 with ICL to convert regulatory descriptions into SUMO-compatible XML files. Expanding on this,

TARGET [86] introduces a multi-stage prompting by using GPT-4 that parses traffic rules into a DSL of CARLA using CoT and ICL. A rule-to-script generator then produces a scenario script. Petrovic et al. [91] extend this direction by processing ADAS test topologies and standardization documents. The test topology is converted into a metamodel that includes elements such as the environment, sensor, and actuator configurations. Standardization documents are parsed into Object Constraint Language (OCL) using LLMs. Based on the combined metamodel, OCL constraints, and a specific test description, an LLM (e.g., GPT-4 or Llama 3) is used to generate DSL test scenarios, which are then simulated in CARLA. In a more data-driven approach, SoVAR [89] reconstructs crash scenarios from NHTSA [73] reports by extracting structured attributes with GPT-4 and generating trajectories and simulation scripts via constraint solving, producing LGSVL [88]-compatible test scenarios via API calls focused on realism. In contrast, LeGEND [90] follows a top-down approach: it abstracts reports into functional scenarios, transforms them into logical DSL representations via a two-stage GPT-4 pipeline, and applies multi-objective search to generate diverse and critical scenarios.

Text2Scenario [92] introduces a standardized hierarchical scenario repository based on the safety of the Intended Functionality (SOTIF) framework and applies multi-stage prompting (CoT, ICL, SC) with GPT-4 to generate logical scenarios from free-form descriptions. The resulting logical scenario is then converted into the OpenScenario format through code and simulated in CARLA. Finally, Zhou et al. [93] focus on lane-keeping systems by using Llama 3.1 and prompt templates to extract scene elements from UNECE R157[7] -aligned descriptions. These descriptions are structured and converted into OpenScenario DSL files using OpenXOntology[8] and OpenDRIVE[9], then simulated in the VTD[10] simulation environment.

While LLM-based frameworks effectively generate ADAS test scenarios from crash reports and regulations, they often overemphasize rare edge cases [89], [90], neglecting common driving scenarios that are essential for broader testing. Currently, we are missing the incorporation of routine test cases and utilizing real-world maps from OSM or SUMO to enhance the scenario diversity and fidelity.

Metrics: Scenario generation for ADAS is commonly evaluated in three ways: (I) *Feasibility* is measured using compile error rate, runtime error rate, and execution rate to assess robustness [91], [92]; (II) *Accuracy* is based on semantic alignment between generated scenarios and input descriptions, often judged by human-labeled matching [89], [92]; (III) *Efficiency*, instead, is measured by generation time or speed-up over manual scripting [91].

C. LLM-based Scenario Analysis

Recent research has explored the use of LLMs as a scenario analysis tool and method. A key challenge is that LLMs are primarily designed to process natural language input, whereas driving scenarios are typically defined using structured data formats, such as scripts in domain-specific language (DSL) or sensor outputs with predefined syntax and semantics. This creates a mismatch between how the scenario's information is represented and how LLMs operate. Therefore, bridging this gap is critical to enable effective interpretation of driving scenarios using language models. We classify the existing works into three key areas and list representative works in Table III.

1) Question Answering (QA): Applying LLMs to scenario analysis for AD requires domain-specific knowledge, which general-purpose pre-trained models may lack. To bridge this gap, fine-tuning with tailored datasets is essential. QA datasets describing driving scenarios help LLMs interpret structured driving contexts, and support downstream tasks like trajectory planning and decision-making.

A notable example is [99], where the authors automate the generation of QA datasets with driving scenarios using GPT-3.5. With a structured language generator, they convert vectorized scenario data from their in-house dataset, including agents' positions, speed, and distance, into natural language. With ICL and pre-defined driving rules, their model generates diverse, context-aware QA pairs to reflect realistic driving situations. However, the QA dataset of [99] focuses primarily on perception and prediction.

Metrics: In [99], the evaluation of QA datasets is performed with (I) *LLM-based QA grading*, relying on GPT-3.5 to assign a score based on the answer's quality, and with (II) *human grading*, which relies on human annotations to assess a subset of QA samples and evaluate the LLMs' results.

2) Scenario Understanding: Here, the LLM processes structured sensor or simulator data, such as agent states, road layouts, and traffic signals, to support tasks like scenario captioning (concise descriptions) and reasoning (coherent narratives capturing intent and context).

The SenseRAG [101] introduces a RAG-based framework from the DLR urban traffic dataset [100] for scenario understanding. They use a VLM to generate traffic condition descriptions into textual descriptions, which are then mapped to a structured database, including additional structured information with weather, city, and traffic participants. Using CoT prompting and SQL query generation, GPT-4 retrieves and reasons over the data to refine perception and enhance trajectory prediction.

Metrics: The evaluation of scenario understanding approaches is primarily based on (I) *trajectory prediction metrics* (e.g., ADE and FDE), which assess the accuracy of the future trajectory predictions made by the LLMs [101].

3) Scenario Evaluation: Recent work demonstrates how LLMs can support the evaluation of driving scenarios, by reasoning over structured simulation data or scenario images converted into natural language. This includes the evaluation

---

[7]https://unece.org/transport/documents/2021/03/standards/un-regulation-no-157-automated-lane-keeping-systems-alks

[8]https://www.asam.net/standards/asam-openxontology/

[9]https://www.asam.net/standards/detail/opendrive/maps

[10]https://hexagon.com/de/products/virtual-test-drive

TABLE III
SUMMARY OF SCENARIO-ANALYSIS STUDIES USING LARGE LANGUAGE MODELS.

Category	Input				Model	Technique[1]	Focus	Paper
	Scenario Elements	Elements Narration	Dataset	Database				
Question Answering (QA) (Sec. III-C1)	Road, Ego, NPC Vehicles, Pedestrians	Language Generator	In-house		GPT-3.5	ICL	Driving QA	Chen et al. [99]
Scenario Understanding (Sec. III-C2)	Road, Weather, Ego, Traffic Light, NPC Vehicles	LLaVA, Language Parsing	DLR UT [100]	Traffic Condition Structured Data	GPT-4	CoT RAG	Reasoning	SenseRAG [101]
Scenario Evaluation (Sec. III-C3)	Road, Weather, Traffic Sign, NPC Vehicles	OWL-ViT, Language Parsing	CARLA		text-davinci-003	CoT ICL	Anomaly Detection	Elhafsi et al. [102]
	Road, Weather, Ego, NPC Vehicles	Vector Parse	DeepScenario [103]		GPT-3.5 LLaMA2-13B Mistral-7B	CP ICL	Realism	Reality Bites [104]
	Road, Ego, NPC Vehicles	Cartesian Parser, Ego Parser	CommonRoad [49]		GPT-4o Gemini-1.5Pro Deepseek-V3	CP CoT ICL	Safety-Criticality	Gao et al. [105]
	Road, Ego, Traffic Light, NPC Vehicles	Not Specified	CARLA	Interview Data	GPT-4o	CoT RAG	Driving Styles	You et al. [106]

[1] Techniques: CoT = Chain-of-Thought prompting; ICL = In-Context Learning; CP = Contextual Prompting; RAG = Retrieval-Augmented Generation.

of anomaly detection, scenario realism, safety-criticality, and driving behavior.

Elhafsi et al. [102] detect semantic scenario anomalies using LLMs. Their scenarios are evaluated using OpenAI's text-davinci-003, which is prompted with CoT and ICL. Reality Bites [104] is one of the first works to evaluate the reasoning ability of LLMs in assessing scenario realism. It transforms XML-formatted DeepScenario [103] data into natural language and uses ICL prompting with models like GPT-3.5, Llama2-13B, and Mistral-7B to judge the alignment with realistic driving conditions. Gao et al. [105] propose a framework to analyze safety-criticality in driving scenarios from the CommonRoad [49] environment. They convert structured scenario data into natural language and prompt LLMs via CP, CoT, and ICL to evaluate the safety-criticality of the scenario and infer the risk level of the agent. Also, they generate safety-critical scenarios by modifying the trajectories of identified adversarial vehicles. Meanwhile, You et al. [106] focus on holistic driving assessment, converting interview and simulation data into a structured knowledge database for RAG. In their framework, GPT-4o classifies driving styles (cautious, aggressive) and performance levels based on aggregated context, including scenario-level information like weather, ego vehicle data, and surrounding traffic participants.

Overall, LLM-based scenario evaluation still depends on token-heavy prompting and handcrafted prompts. Emerging reasoning models like OpenAI o1 and DeepSeek-R1 may enable more efficient, zero-shot approaches.

Metrics: Scenario evaluation uses task-specific accuracy metrics: (I) *Realism* via robustness scores that measure how consistently the LLMs performs when the same scenario is perturbed (e.g., changes in agent behavior). Higher robustness indicates better generalization to realistic variations [102], [104]. (II) *Criticality* via accuracy in identifying high-risk agents that are likely to lead to collisions [105]; (III) *Behavior classification* via evaluating how LLMs classifies driving behavior into the predefined styles like aggressive, cautious, or predefined performance levels [106].

IV. VISION LANGUAGE MODELS (VLMS)

This section introduces VLMs, summarizes key adaptation techniques, and surveys VLM-based scenario generation for safety-critical, real-world, datasets, and ADAS testing applications. Additionally, it explores how VLMs support scenario analysis activities such as visual question answering (VQA), scene understanding, benchmark & dataset, and risk assessment.

A. Development of VLMs

1) Evolution of VLMs: The Vision Transformer (ViT) [15] extended the transformer architecture from NLP to computer vision by dividing an image into fixed-size patches. This enabled embedding and image as a token and processing the sequence of tokens with a standard transformer encoder. This success inspired researchers to combine visual and textual modalities, leading to the development of VLMs, which now can jointly process images and text at the same time. A milestone was the development of CLIP (Contrastive Language–Image Pretraining) [27], which was trained on hundreds of millions of image–text pairs using a contrastive learning objective, enabling effective zero-shot performance without task-specific supervision. ALIGN [107] scaled this approach to billions of noisy web-crawled pairs. BLIP [108] unified multiple tasks with captioning and retrieval into a single training framework. Flamingo [109] introduced few-shot multimodal prompting with frozen backbones and cross-attention layers for rapid adaptation.

By leveraging VLMs's ability to jointly reason over images and text, researchers have explored new concepts for autonomous driving taks. As summarized in recent surveys [55], [56], VLMs enable interpretable and adaptable systems that support open-ended interaction, improve generalization to unseen scenarios, and facilitate multimodal reasoning. These advancements mark a shift toward more intelligent and explainable autonomous vehicles, laying the groundwork for safer and more human-aligned driving agents.
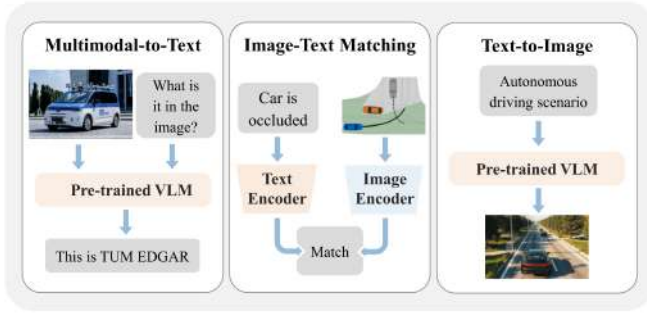
Fig. 5. Overview of main applications of VLMs for autonomous driving.

VLMs are broadly categorized into three major application domains for ADs, as illustrated in Figure 5:

(I) Multimodal understanding which jointly processes images and text, such as image captioning and visual question answering. Flamingo [109] is notable for its few-shot VQA, while BLIP [108] supports captioning, retrieval, and question-answering in a unified model.

(II) Image–text matching which involves assessing semantic alignment between an image and a caption, and is critical in large-scale visual search. ALIGN [107] and CLIP [27] set the benchmark in contrastive pretraining for retrieval performance.

(III) Text-to-image generation which involves synthesizing novel visuals from natural language prompts. DALL-E [110] pioneered this area by using a transformer to produce coherent and diverse images from descriptive text.

2) Adaptation Techniques for VLMs: VLMs are typically pre-trained and subsequently adapted for downstream tasks. Beyond standard prompt engineering the following several adaptation strategies are commonly employed in the context of scenario generation and analysis in ADs.

Modality alignment modules: These are additional modules which are trainable and serve to transform visual or structured features into a format compatible with the language model. Common approaches include:

(I) *Q-Former:* A transformer with learnable queries that aligns image features with the language model input space via cross-attention (e.g., BLIP-2 [111]).

(II) *Cross-attention:* Used to resample variable-length image or video tokens into a fixed-size latent representation, enabling consistent language interaction (e.g., Flamingo [109]).

(III) *MLP mapping:* A linear or multi-layer perceptron (MLP) projects vision encoder outputs to match the dimensionality required by the language model [112], [113].

(IV) *Structure-aware encoder (Prior Tokenizer):* A perception-aware module that encodes structured detection outputs, such as semantic attributes, into token embeddings for downstream reasoning. For example, Reason2Drive [114] introduces a module called Prior Tokenizer to fuse region features with object-level semantics.

Fine-Tuning: Fine-tuning techniques train VLMs on datasets of instruction–response pairs involving both visual and textual inputs to improve their ability to follow multimodal instructions. Two main strategies are used:

(I) *Full fine-tuning (Fft):* All model weights are updated on the target dataset. It typically yields the highest task performance but incurs high computational costs and risks of overfitting. Several reviewed works adopt full fine-tuning for smaller VLMs, striking a practical balance between effectiveness and efficiency [112], [115].

(II) *Parameter-Efficient Fine-Tuning (PEFT):* PEFT methods enable adaptation by updating only a small number of additional parameters. One of the common methods is Low-rank matrices (LoRA), which are injected into attention or feed-forward layers, enable efficient adaptation with minimal parameter overhead [94].

B. VLM-based Scenario Generation

This subsection reviews how VLMs are used to generate driving scenarios by leveraging their understanding of visual and textual inputs. We group recent works into four categories and display them in (Table IV):

1) Safety-critical Scenario Generation: Safety-critical scenario generation is a rapidly advancing application of VLMs in autonomous driving. It enables the synthesis of rare but relevant situations that are essential for evaluating system robustness. By combining visual perception with semantic understanding, VLMs can detect failures and generate targeted, interpretable scenarios.

Recent frameworks such as CurricuVLM [116] illustrate the potential of VLMs. CurricuVLM integrates VLMs like LLaVA into an online curriculum learning loop, where BEV images and task descriptions are analyzed to identify safety-critical events, while GPT-4o uncovers behavioral weaknesses through batch-wise pattern analysis. These insights guide a pre-trained DenseTNT model in generating personalized agent trajectories, with scenario selection dynamically adapted through reinforcement learning.

CurricuVLM employs general-purpose pre-trained VLMs, thus their performance in identifying safety-critical agents is limited. Future work could explore combining these frameworks with safety-aware, fine-tuned VLMs and incorporating temporal and multi-sensor contexts to improve reliability.

Metrics: The evaluation here only focuses on *safety-critical metrics* such as collision rate and driving distance [116], to evaluate the severity and effectiveness of the generated scenarios.

2) Real-World Scenario Synthesis: VLMs offer new opportunities for realistic driving scenario generation by combining natural language understanding with visual modalities such as scenario images for generation or interpretation, enabling the creation of highly realistic traffic scenes.

OmniTester [117] proposes a framework with LLM and VLM to create realistic and diverse traffic scenarios in SUMO. User inputs and context from RAG with external knowledge and OSM map library are processed via GPT-4 to generate SUMO scenario scripts. A VLM GPT-4V analyzes the generated scenario using images and code, providing feedback in natural language. Then, the GPT-4 evaluator

TABLE IV
SUMMARY OF SCENARIO-GENERATION STUDIES USING VISION LANGUAGE MODELS.

Category	Input					Model	Technique[1]	Simulator	Output[2]	Paper
	Text	Image	View Type	Dataset	Database					
Safety-critical Scenario (Section IV-B1)	✓	✓	BEV of Metadrive	Waymo Open		GPT-4o LLaVA	CoT	Metadrive	[trajectory]	CurricuVLM [116]
Real-world Scenario (Section IV-B2)	✓	✓		SUMO	Road Network	GPT-4 GPT-4V	CoT RAG	SUMO	[script]	OmniTester [117]
Dataset (Section IV-B3)	✓			In-house		DALL-E2	ICL		[image]	WEDGE [118]
ADAS Test Scenario (Section IV-B4)		✓	Real FPV	CDD [119]		GPT-4o	ICL	CARLA	[script]	Miao et al. [120]
	✓	✓	BEV of Sketch	nuScenes [121]	NHTSA	GPT-4o	CoT ICL	Metadrive BeamNG	[script]	TRACE [122]

[1] Techniques: CoT = Chain-of-Thought prompting; ICL = In-Context Learning; RAG = Retrieval-Augmented Generation.

[2] Output: [icon] image, [icon] Trajectory, [icon] Scenario script.

compares this feedback against the intended description to enhance scenario generation.

Beyond map-based scenario synthesis, we are currently missing replaying real-world logs for more realistic behavior and reconstructing crash scenarios for risk-focused scenario testing. VLMs can enhance these by captioning generated scenarios and using the descriptions as feedback to iteratively improve generation accuracy.

Metrics: In OmniTester [117], the evaluation focuses on two key aspects. (I) *Controllability* measures how well the system adheres to user-specified requirements, including scene type, the number of lanes or vehicles, and whether the generated files are valid for execution in SUMO. (II) *Diversity* captures the range and variability of generated scenarios, evaluated through statistical analysis of lane counts, edge counts, route lengths, and vehicle numbers.

3) Dataset Generation: A key application of VLMs is text-to-image generation to build tailored driving datasets, particularly to improve perception systems under diverse conditions.

WEDGE [118] showcases the use of VLMs, specifically DALL-E 2, to synthesize images depicting 16 diverse and extreme weather conditions relevant to autonomous driving. Their dataset includes manually annotated 2D bounding boxes and is used to fine-tune object detectors. When evaluated on the real-world dataset, object detectors trained on WEDGE exhibit improved detection performance, highlighting the potential of VLM-generated data for enhancing perception robustness in adverse conditions.

Currently we are missing a hybrid training that combines real and synthetic data, as well as the targeted generation of rare or high-risk scenarios, such as crashes, occlusions, or anomalies, to build synthetic datasets for safety-critical testing and anomaly detection.

Metrics: (I) *Perception metrics* such as average precision for object classes (e.g., truck, car); and (II) *Image similarity metrics* like peak signal-to-noise ratio, structural similarity index, and root mean squared error, used to assess visual fidelity between synthetic and real-world images [118].

4) Generation of ADAS Test Scenarios: VLMs extend ADAS scenario generation by grounding language in visual content, enabling semantically rich and visually faithful reconstructions of complex driving events. This facilitates realistic reconstructions from sources like crash reports or driving logs by integrating language with visual context.

In [120], the authors present a fully automated pipeline that transforms sample frames of dashcam crash video from the Car Crash Dataset [119] into simulation scenarios for ADAS testing. Using GPT-4o with ICL, the system generates SCENIC scripts for CARLA, while a second GPT-4o compares real and simulated video frames based on predefined behavior features, enabling iterative refinement through visual feedback. Similarly, TRACE [122] reconstructs realistic ADAS test scenarios from unstructured multimodal crash reports, including textual summaries and visual sketches. It uses GPT-4o with ICL and CoT to extract road types and environmental details from sketches. An LLM, built on GPT and augmented with trajectory data from nuScenes [121], generates realistic vehicle paths. These components are transformed into a DSL-based scenario compatible with simulators like MetaDrive using a rule-based encoder.

Currently, we are missing online interactive scenario editing, where users modify scenes by sketching or annotating video frames, and VLMs dynamically update the simulation code. This would enable human-in-the-loop control and more flexible scenario refinement.

Metrics: (I) *Fidelity*, measured by human ratings on trajectory similarity and scene consistency [120], [122]; (II) *Accuracy*, including environment attribute matching, road network parameters, and actor classification [122]; (III) *Efficiency*, assessed by scenario generation time and success rate of fully automated pipelines [120].

C. VLM-based Scenario Analysis

The current progress in sceneratio generation with VLMs is quite at the beginning, but VLMs have shown already big promises for scenario analysis in AD. Examples include NuScenes-QA [128] for Visual Question Answering (VQA), where a VLM answers natural language questions grounded in driving scenes to support scenario analysis; NuPrompt [157] for language-guided tracking and prediction, and Refer-KITTI [158] for multi-object referring tracking tasks. However, these models are not considered foundation models because they do not utilize fully pre-trained foundation

TABLE V
SUMMARY OF SCENARIO-ANALYSIS STUDIES USING VISION LANGUAGE MODELS.

Category	Input			Model			Technique²	Focus	Paper
	Context	Image¹	Dataset	VLM	LLM	Role			
Visual Question Answering (VQA) (Section IV-C1)		Multi-view	nuScenes	BLIP2 InstructBLIP2 MiniGPT4	GPT-4	VLM: BEV Feature Extraction LLM: QA Execution	Zero-shot	Perception Prediction	Talk2BEV [123]
	✓	Multi-view	nuScenes	ViT+OPT	GPT-4	VLM: VQA Execution LLM: QA Generation	MLP Fft	Perception	NuScenes-MQA [112]
	✓	Multi-view	nuScenes	EVA-02-L + Llama2-7B	GPT-4	VLM: VQA Execution LLM: QA Augmentation	MLP Q-Former Fft	Counterfactual Reasoning	OmniDrive [113]
	✓	FPV	nuScenes Waymo Open Once [124]	FlanT5-XL + Vicuna-7B	GPT-4	VLM: VQA Execution LLM: QA Augmentation	Q-Former Tokenizer LoRA	Perception Prediction Reasoning	Reason2Drive [114]
	✓	Multi-view	nuScenes	InterVL2.5-8B	GPT-4o	VLM: VQA Execution LLM: QA Generation	LoRA	Perception Prediction Planning	DriveLMM-o1 [125]
	✓	FPV	DriveLM [126] LingoQA [127] NuScenes-QA [128]	Qwen2-VL-7B Qwen2-VL-72B GPT-4o	GPT-4o	VLM: VQA Execution LLM: Multi-Choice QA	Zero-shot	Perception Prediction Planning	AutoDrive-QA [129]
Scene Understanding (Section IV-C2)	✓	Multi-view	Waymo Open	CLIP			Zero-shot	Tagging	Najibi et al. [130]
	✓	Multi-view	SemanticKITTI [131]	Grounding DINO + SAM	GPT-3.5	VLM: Object Grounding LLM: Narrative Generation	Zero-shot	Tagging	OpenAnnotate3D [132]
	✓	FPV	Cityscapes [133] CamVid [134] CARLA	ImageGPT			Zero-shot	Tagging	Kou et al. [135]
	✓	Multi-view	DriveLM	ViT-B/32 + T5-Base/Large			MLP Fft/LoRA	Tagging	EM-VLM4AD [115]
	✓	FPV	FARS	GPT-4V LLaVA-13B	Llama2-13B Zephyr-7b-α	VLM: Scene Captioning LLM: Risk Assessment	Zero-shot	Captioning	Zarzà et al. [136]
	✓	BEV	WOMD [46]	GPT-4V			Zero-shot	Captioning	Zheng et al. [137]
	✓	FPV	BDD100K [138]	Multiple VLMs			Zero-shot	Tagging Reasoning	Rivera et al. [139]
	✓	FPV	nuScenes BDD-X [140] CDD [119]	GPT-4V			Zero-shot	Reasoning	Wen et al. [141]
	✓	Multi-view	MAPLM-QA [142]	ViLA			Zero-shot	Reasoning	Keskar et al. [143]
Benchmark & Dataset (Section IV-C3)	✓	FPV	DriveLM	Multiple VLMs	GPT-4o	VLM: VQA Execution LLM: Answer Evaluation	Zero-shot	Robustness	DriveBench [144]
	✓	Multi-view	nuScenes	ViT/V2-99 + LLaVA-1.5-7B			Tokenizer MLP LoRA	3D Grounding	NuGrounding [145]
	✓	FPV	nuScenes CARLA	BLIP2			LoRA	GVQA	DriveLM [126]
	✓	FPV	CODA [146]	LLaVA-llama-3-8B GPT-4o	GPT-4	VLM: Scene Captioning LLM: Caption Evaluation	LoRA	Corner Cases	CODA-LM [147]
	✓	FPV	In-house	GPT-4o			CP, ICL, CoT	ADAS-LKA	OpenLKA [148]
Risk Assessment (Section IV-C4)	✓	Multi-view	In-house	GPT-4V			CP CoT	Risk Scoring	Hwang et.al [149]
	✓	FPV	DAD [150]	Flamingo	GPT-3.5		Zero-shot	Hazard Explanation	Latte [151]
	✓	FPV	CARLA	DINOv2 OWLV2+SAM2 GPT-4o			CP	Anomaly Detection	Ronecker et.al [152]
	✓	Multi-view	CARLA	InternViT	Interlm2-chat	VLM: Video Extraction LLM: Narrative Generation	MLP QLoRA CP, CoT	Situation Awareness Reasoning	Think-Driver [153]
	✓	Partially occluded BEV	CARLA	Llama3.2-11B LLaVA-1.6-7B Qwen2-VL-7B			LoRA CP	Uncertainty Scoring	Lee et.al [154]
	✓	FPV	BDD100K	Qwen2-VL7B			LoRA	Hazard Detection	INSIGHT [155]
	✓	FPV	OpenLKA [148]	Qwen2.5-VL-3B Qwen2.5-VL-7B			LoRA	LKA Failures Prediction	LKAlert [156]

¹ Image: FPV = First Person View; BEV = Bird Eye View.
² Techniques: Fft = Full fine-tuning; CP = Contextual Prompting; ICL = In-Context Learning; CoT = Chain-of-Thought prompting; Tokenizer = Prior Tokenizer; MLP = Multi-Layer Perceptron mapping.

architectures. Rather, they construct task-oriented frameworks based on LLM backbone components.

In this section, we focus on foundation VLMs pre-trained on large-scale, diverse image–text datasets with cross-domain generalization. We examine their potential to improve transferability, explainability, and efficiency in analyzing complex AD scenarios. We structure our discussion around four key application areas and show their various techniques and applications in Table V.

1) Visual Question Answering (VQA): VQA datasets for autonomous driving pair visual inputs with natural language queries, to evaluate scene understanding across tasks such as perception, prediction, and planning. While recent works have proposed VQA datasets, some remain conceptual or require human reasoning in their creation, without leveraging LLMs for generation or VLMs for action execution. This section focuses on VQA-based scenario analysis methods that leverage VLMs to analyze complex driving environments.

Early efforts began by enriching existing scene

representations with the *perception* task. Talk2BEV [123] uses a perception stack to generate BEV maps by fusing multi-view images and LiDAR, then applies BLIP-2 to augment these maps with object-level language descriptions. These descriptions are passed to GPT-4 with CoT prompting to answer spatial and semantic queries, enabling zero-shot VQA with annotated QA pairs focusing on perception and prediction. Similarly, NuScenes-MQA [112] uses GPT-4 to automatically generate diverse question templates within the Markup-QA scheme. The authors fully fine-tune a VLM that combines a CLIP-pre-trained ViT as a visual encoder, and OPT as a language model, using an MLP to align multi-camera visual features with text. This setup enables joint evaluation of caption generation and visual question answering in driving scenarios for perception.

Later works moved toward more advanced *reasoning* tasks. OmniDrive [113] introduces the first 3D VQA dataset for counterfactual reasoning in autonomous driving, evaluating VLMs with frozen EVA-02-L and Llama2–7B backbones, and using either an MLP projector (Omni-L) or a Q-Former (Omni-Q) as trainable modality bridges. Reason2Drive [114] presents a video–text VQA dataset composed of sequential images from nuScenes, Waymo, and ONCE [124], covering tasks in perception, prediction, and reasoning. The authors fine-tune a VLM consisting of FlanT5-XL and Vicuna-7B by using LoRA, leveraging a prior tokenizer and an instructed vision decoder. A Q-Former module is employed to jointly predict answers and perceptual cues.

Recent efforts in VQA-based scenario analysis focus on advancing multimodal reasoning and evaluation across *perception, prediction*, and *planning* tasks in autonomous driving. DriveLMM-o1 [125] introduces a step-by-step reasoning dataset based on nuScenes, incorporating both images and LiDAR points into the QA context. Their QA pairs are initially generated using GPT-4 and subsequently refined through human annotation. The authors fine-tune InternVL2.5-8B using LoRA, demonstrating improved performance on reasoning and final answer accuracy across perception, prediction, and planning. AutoDrive-QA [129] converts open-ended QA pairs from DriveLM [126], LingoQA [127], and NuScenes-QA [128] into multiple-choice questions using GPT-4o, adding distractors, which are plausible but incorrect answer choices designed to reflect realistic domain-specific errors, to simulate realistic errors. This forms a standardized benchmark to evaluate pre-trained VLMs across key scenario analysis tasks across perception, prediction, and planning.

Despite these advances, most of the current VQAs overlook traffic rules and real-world driving conventions. Future work should incorporate rule-aware QA, grounded in road semantics (e.g., right-of-way rules and road signal compliance), to enable more realistic and safety-relevant scenario reasoning.

Metrics: (I) *Answer Accuracy*, based on binary or multiple-choice correctness in response to perception, planning, and reasoning questions [112], [123], [125]; and (II) *Language Generation Quality*, measured using BLEU, CIDEr, METEOR, and ROUGE-L scores to evaluate caption and reasoning fluency [112], [114];

2) Scene Understanding: VLMs are heavily used to interpret complex driving scenarios.

Recent works have leveraged VLMs for *scene tagging*, which represents the most basic level of scene understanding, involving binary or categorical assignments. *Scene tagging* assigns predefined labels at either the scene level (e.g., to analyze the weather conditions), or at pixel level (semantic segmentation) to characterize visual content for downstream tasks. Najibi et al. [130] leverage a pre-trained VLM to perform zero-shot scene tagging on camera images, assigning semantic labels that are projected onto LiDAR points. These labels guide the generation of 3D pseudo-labels, which are then used to train a 3D object detector without human annotations. OpenAnnotate3D [132] introduces an auto-labeling system for multi-modal 3D data, using GPT-3.5 for interpreting natural language scene descriptions and a VLM with Grounding DINO and SAM for generating dense 2D masks, which are fused spatio-temporally and projected into 3D annotations. Kou et al. [135] propose a framework to enhance VLMs for street scene semantic understanding. They use a pre-trained ImageGPT to extract semantic features from First Person View (FPV) images, and apply a *posterior optimized trajectory* strategy to train a lightweight perception head that maps the semantic features to pixel-wise semantic segmentation masks. EM-VLM4AD [115] proposes a lightweight VLM trained on the dataset from DriveLM [126] with a primary focus on scenario tagging. It uses a ViT image encoder and explores two adaptation strategies: full fine-tuning of T5-base and LoRA-based tuning of T5-large. The model is benchmarked against baselines in terms of parameter count, Floating Point Operations Per Second (FLOPs), and memory usage, showcasing strong efficiency for deployment in resource-constrained settings.

Building on scene tagging, recent efforts have advanced toward the intermediate-level task of *scene captioning*, which bridges perception and language by generating open-form descriptions. *Scene captioning* generates concise natural language descriptions of visible elements. Zarzà et al. [136] propose a framework using structured inputs with principal component analysis, and adopt Llama2-13B with CoT and CP to assess the risks in a scenario, suggesting driving adaptations. They test their framework with the FARS dataset[11]. Additionally, they leverage a VLM, specifically LLaVA-13B with CP, to perform image-based scenario captioning, enhancing scene understanding through natural language descriptions. Zheng et al. [137] introduce a context-aware motion prediction framework using VLMs. They employ GPT-4V to extract traffic context from a *transportation context map*. They combine vector map data and historical trajectories, and feed the generated scenario description into a motion transformer to improve trajectory prediction.

Several studies address the most advanced form of scene understanding: *scene reasoning*, which requires interpreting interactions, causality, and abstract situational context. *Scene reasoning* interprets relationships and interactions among

---

[11]https://www.nhtsa.gov/research-data/fatality-analysis-reporting-system-fars

agents while producing coherent narratives that capture intent, causality, and situational context. Rivera et al. [139] propose a scalable pipeline for traffic scene classification using off-the-shelf VLMs such as GPT-4V, LLaVA, and CogAgent-VQA [159]. These models are evaluated zero-shot to reason about predefined scenario elements, such as lane markings and vehicle maneuvers, using self-developed and the BDD100K [138] datasets. Wen et al. [141] explore GPT-4V's zero-shot capability for road scene interpretation from dashcam footages, evaluating the model on object detection, scene captioning, VQA, and causal reasoning, while highlighting its potential and limitations for autonomous driving. Keskar et al. [143] evaluate NVIDIA's ViLA on the MAPLM-QA benchmark for traffic scene understanding. Using contextual prompting, they assess ViLA on multiple-choice VQA tasks, including lane counting, intersection detection, scene classification, and point cloud quality assessment. ViLA shows strong performance on high-level VQA tasks but struggles with fine-grained spatial reasoning.

Metrics: The evaluation of VLM-based scenario understanding uses the following metrics. (I) *Scene tagging*: common metrics are *Intersection-over-Union* and mean F1 score [132], [135], which quantify the segmentation performance, and the BLEU-4, METEOR, ROUGE-L, and CIDEr metrics to assess the textual tagging outputs [115]. Additionally, mean average precision is used for evaluating 3D object detection from pseudo-labels [130]. (II) *Scene captioning:* human or LLM-based evaluation of causal coherence [136], and functional evaluation via intention-prediction accuracy, confusion matrices, and ablation studies on prompt designs [137]. (III) *Scene reasoning:* the evaluation includes attribute classification in zero-shot setups [139], task-specific accuracy on structured VQA tasks such as lane counting and intersection detection [143], and rubric-based assessments of causal reasoning and multimodal consistency [141].

3) Benchmarks & Datasets: To support the development and evaluation of VLMs in autonomous driving, recent efforts have introduced specialized *benchmarks* and curated *datasets* covering key tasks such as perception, prediction, planning, and scenario reasoning under real-world and safety-critical conditions.

Aiming for a standardized evaluation, several works present *benchmarks* aligned with diverse driving scenarios. DriveBench [144] introduces a benchmark for evaluating scenario reasoning across multiple driving tasks. It extends the VQA dataset from DriveLM [126] and adds diverse visual corruption categories to assess the model's robustness. Using this benchmark, the authors evaluate the robustness of a range of pre-trained and fine-tuned VLMs (e.g., GPT-4o, Qwen2-VL, Dolphins) under clean, corrupted, and text-only conditions. GPT-4o is further employed as an automatic evaluator for open-ended answers. nuGrounding [145] proposes the first 3D visual grounding benchmark with human-annotated object grounding based on nuScenes. The authors fine-tune a VLM, LLaVA-1.5, using LoRA, with ViT or V2-99 as the visual

encoder. To incorporate 3D understanding, they extract BEV features via a BEV-based detector, map them into the LLM adapter, and fuse them with VLM outputs through a query fuser for accurate object detection and localization.

Complementing these benchmarks, other works provide high-quality *datasets* to train and adapt VLMs to complex driving environments. DriveLM [126] introduces a graph-structured visual question answering (GVQA) dataset, using human-curated QA graphs from nuScenes and rule-based annotations from CARLA. A BLIP-2-based VLM is fine-tuned with LoRA and guided by graph-based question prompting to enable zero-shot interpretable scenario reasoning across perception, prediction, and planning. CODA-LM [147] introduces a corner-case image-text dataset derived from the CODA dataset [146]. The authors use GPT-4V to generate multi-task captions spanning perception, prediction, and planning for each image. These captions are then evaluated and refined using GPT-4. After constructing the dataset, they fine-tune a LLaVA-llama-3-8B model to enhance vision language understanding in corner-case driving scenarios. OpenLKA [148] introduces a large-scale, real-world dataset for Lane Keeping Assist under diverse driving conditions. GPT-4o is used in conjunction with CP, CoT, and ICL to generate structured scene annotations that describe lane quality, weather, and traffic context.

However, the current benchmarks and datasets lack realism and diversity. For example, DriveBench exposes the VLM's vulnerability to corruption, suggesting the need for more realistic disturbances (e.g., occlusions, night-time). CODA-LM relies on filtered GPT captions, underscoring the gap in real-world edge-case coverage.

Metrics: The evaluation of benchmarks focuses on (I) *robustness*, which measures the VQA performance under clean, corrupted, and text-only inputs [144]. Another used metric is (II) *grounding precision*, which evaluates 3D mean Average Precision to evaluate 3D object detection performance and alignment accuracy for spatial localization and multimodal fusion [145]. The evaluation of datasets focuses on (III) *answer fidelity*, to assess the QA correctness and consistency via graph-guided prompts [126]; (IV) *caption quality*, which uses BLEU-4, METEOR, ROUGE-L, and CIDEr [147]; (V) *annotation coverage*, which measures attribute diversity and alignment with telemetry data [148].

4) Risk Assessment: VLMs are increasingly applied to autonomous driving risk assessment, addressing tasks like hazard detection, uncertainty estimation, and failure prediction. Recent approaches leverage both prompting and fine-tuning and use diverse visual inputs, including BEV maps, multi-view images, and segmentation masks. These methods aim to improve safety through interpretable reasoning and context-aware decision support.

Recent advances in VLM-based risk assessment have explored prompting techniques for risk analysis. Hwang et al. [149] utilize GPT-4V in a zero-shot setting for risk scoring in street-crossing scenarios. The model receives structured visual inputs—bounding boxes, segmentation masks, and optical flow—alongside contextual prompts

formulated using CoT. Instead of directly processing raw images, GPT-4V reasons over augmented visual features to assess safety levels and provide natural language justifications. Similarly, LATTE [151] introduces a real-time hazard detection framework that utilizes off-the-shelf computer vision modules and three lightweight attention modules for spatial reasoning, temporal modeling, and risk prediction. Upon hazard detection, Flamingo and GPT-3.5 are triggered to generate scene captions and verbal explanations. The system operates in a zero-shot manner by leveraging contextual prompting for situational reasoning. For anomaly object detection, Ronecker et al. [152] proposed both patch-based and instance-based embedding methods using vision foundation models, evaluated on a CARLA-based dataset. They leverage the zero-shot capabilities of DINOv2 for visual embeddings and combine OWLv2 with SAM2 for object-level instance segmentation. Their instance-based approach achieves performance comparable to, or slightly better than, GPT-4o using contextual prompting.

Think-Driver [153] proposes a VLM that uses multi-view images to assess perceived traffic conditions and evaluate the risks of current driving maneuvers. It employs multi-view RGB inputs and ego state data, processed by InternViT and InterLM2-chat, respectively. The model is fine-tuned using QLoRA and trained on CoT-style question-answering data that cover scene understanding, hazard reasoning, and action prediction. In consideration of occlusion-aware BEV representations, Lee et al. [154] first investigate the use of VLM for uncertainty prediction in autonomous driving. They construct a dataset from CARLA using BEV images that contain occlusion masks, paired with driving actions and uncertainty scores. Three VLMs are fine-tuned using LoRA to compare their performance under occluded conditions. For hazard detection and explanation, INSIGHT [155] fine-tunes Qwen2-VL-7B via LoRA. Using annotated hazard locations in BDD100K images, the model is trained to localize high-risk regions and generate natural language descriptions. It outperforms several pre-trained VLMs in both spatial localization and interpretability tasks. Finally, LKAlert [156] develops a VLM-based framework for predicting lane-keeping assist failures. It integrates RGB dashcam images, CAN bus signals, and lane segmentation masks from LaneNet. A Qwen2.5-VL model is fine-tuned via LoRA, with lane masks serving as spatial guidance. The model outputs binary alerts and interpretable explanations to enhance safety transparency.

To enable real-world deployment, we must reduce inference latency and resource demands through model compression, efficient prompting, and lightweight VLM architectures optimized for onboard execution in autonomous vehicles.

Metrics: (I) *Risk Scoring:* classification accuracy and human-rated explanation quality [149]; (II) *Hazard and Anomaly Detection:* Evaluated via localization accuracy and interpretability via human evaluation [155], [156], and anomaly object detection accuracy [152] (III) *Risk Forecasting and Failure Prediction:* performance is measured via uncertainty score quality, action accuracy [154], binary failure classification [156], time-to-accident metrics [151].

V. Multimodal Large Language Models (MLLMs)

This section begins with the development of MLLMs, highlighting their architectural evolution and adaptation techniques such as modality bridging and instruction tuning. Then, it covers scenario generation from multimodal input and scenario analysis tasks, including VQA, scene understanding, and risk assessment in AD contexts.

A. Development of MLLMs

1) Evolution of MLLMs: Multimodal Large Language Models extend pretrained LLMs by incorporating one or more perception encoders, enabling the system to process richer sensor inputs. Earlier models like BLIP-2 [111] and Video-LLaMA [160] (discussed in Section IV-A2) use frozen vision backbones connected to LLMs via adapters such as Q-Formers. Recent MLLMs shift to tighter vision–language integration, treating the LLM as a central reasoning engine. MiniGPT-4 [161] and LLaVA [28] align vision encoders (e.g., CLIP) to LLaMA [98] via lightweight projectors and apply instruction tuning. However, they mainly handle natural images, such as typical photographic scenes, and lack support for sensor fusion or spatiotemporal reasoning.

In contrast, models like GPT-4V [162] and GPT-4o integrate proprietary vision encoders and are trained on image–text–audio data. While capable of handling images and videos, they still operate under a natural image paradigm and are not exposed to domain-specific inputs like LiDAR sensor data, BEV maps, or driving logs. Models like Google Gemini [163] and Qwen-VL [164] extend multimodal reasoning with a stronger vision–language integration, supporting open-ended video and chart understanding. While effective for general visual-language tasks, these models fall short in autonomous driving, which requires reasoning over structured inputs like temporal object tracks, BEV layouts, and interaction-aware motion patterns.

To address the unique demands of autonomous driving, recent MLLM architectures have begun incorporating structured, domain-specific modalities such as multi-view video, LiDAR point clouds, and BEV layouts. These additions enable spatial and temporal grounding, allowing LLMs to reason more effectively over complex driving scenes and multi-agent dynamics [165].

2) Adaptation Techniques for MLLMs: While building on techniques from VLMs, MLLMs are adapted to support a broader range of modalities essential for autonomous driving, such as video, LiDAR point clouds, BEV maps, and HD semantic features. As illustrated in Figure 6, these systems typically consist of specialized modality encoders (e.g., BEVFormer [166], CLIP-ViT [167], VoxelNet [168]), projection modules to align multi-modal features (e.g., MLPs, Q-Former, cross-attention), and task-specific training strategies. MLLMs often keep both the perception and language backbones frozen, with adaptation focusing on lightweight bridging and instruction tuning for downstream driving-related tasks. The main adaptation strategies are discussed in the following.
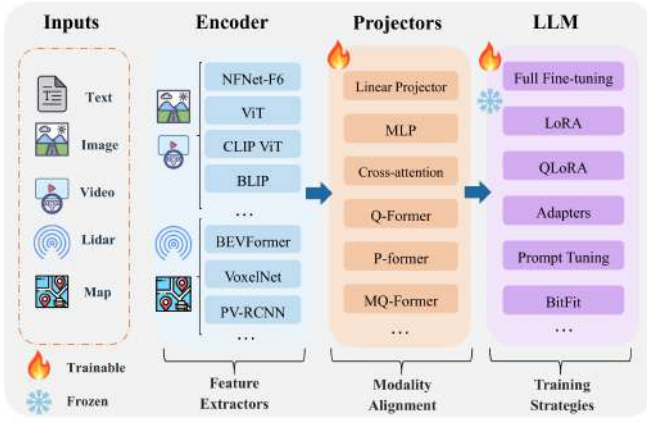
Fig. 6. Overview of adaptation techniques for MLLMs in autonomous driving. **Encoders** extract features from modality-specific inputs. **Projectors** are trainable modules that map features into the LLM's embedding space to enable cross-modal alignment. The **LLM** serves as the reasoning core and can be frozen or trainable, depending on the available resources and the task, using fine-tuning techniques.

Modality alignment modules: These modules serve as a bridge between non-text modalities and the LLM's token space. The main modality alignment modules are:

(1) Linear projector: a single linear (i.e., fully connected) neural layer used to project modality-specific features into the LLM's embedding space. It offers a lightweight mapping strategy and is often used in early-stage VLMs or in combination with pretrained encoders [169], [170].

(2) MLP projection: Projects high-dimensional features from vision or spatial encoders (e.g., ViT, BEVFormer, PointNet) into the LLM's token space. Used in models such as BLIP-2 [111] and driving-centric adapters like P-Adapter [171], which align BEV or LiDAR features for language-based reasoning.

(3) ST-Adapter: A lightweight temporal adapter module is used to extend image-based MLLMs to process sequential video inputs [166], [171]. It enables spatiotemporal modeling without modifying the core LLM weights.

(4) Cross-attention: Uses learnable queries to attend over image or point cloud tokens, enabling multimodal fusion for tasks such as spatial/temporal grounding, semantic alignment, and instruction following [172], [173].

(5) Query transformer: Transformer-based query modules that distill task-relevant embeddings from multi-modal inputs using cross-attention. These modules are applied in BLIP-2 [111], InternDrive [174], and NuInstruct [166] for structured fusion across video, LiDAR, and BEV inputs. In BLIP-2, the module is specifically referred to as Q-Former.

(6) Fusion transformers: Specialized attention blocks designed to integrate features across multiple streams such as BEV maps, multi-view video, or LiDAR point clouds. Modules like BEV-Injection [166] serve as fusion transformers by aligning and injecting multi-modal features (e.g., from images or LiDAR) into a unified BEV representation. These are commonly used in driving-centric MLLMs.

(7) Structure-aware encoder: A module that converts structured perception inputs, such as 3D bounding boxes [168],

scene graphs, or motion trajectories, into token embeddings suitable for language-based reasoning.

Multimodal fine-tuning: Once modality alignment is achieved, an MLLM can be trained to follow task-specific prompts using paired instruction data like VQA. This stage teaches the model to reason over multimodal contexts and produce grounded outputs. Similarly to VLMs, two main strategies are commonly employed to achieve this adaptation:

(1) Parameter-efficient fine-tuning (PEFT): PEFT strategies adapt MLLMs by updating only a small subset of the model's parameters, typically keeping the LLM frozen. While classical PEFT methods such as adapter layers, LoRA, and prompt tuning (discussed below) operate inside the LLM, recent works in autonomous driving often apply PEFT to modality alignment modules [166]. For example, components like ST-Adapters and Q-Formers are trained to bridge visual or spatial inputs to the LLM, enabling task adaptation without modifying the core language model.

Adapter layers: Lightweight trainable modules inserted between the layers of an LLM, typically using a down-projection and up-projection structure. They are used in LLaMA Adapter V2 [175] and InternDrive [174].

LoRA: Applies low-rank updates to attention and feed-forward modules. Frequently used in driving models like DriveGPT4 [169].

PEFT-MA: Fine-tuning only the Modality Alignment (MA) modules (e.g., Q-Former, ST-Adapter), while keeping the LLM's weights frozen.

(2) Full fine-tuning: Full fine-tuning updates all model parameters, including vision encoders, spatial encoders, and the LLM. While this approach typically yields the highest task-specific performance, it is computationally intensive. To reduce the computational cost, some works apply full fine-tuning to smaller models, for example using Qwen2-0.5B [176].

B. MLLM-based Scenario Generation

MLLMs can jointly process a high variety of visual inputs either created by the car or created by humans. The availability of such inputs enables a comprehensive understanding of complex driving environments. Moreover, MLLMs' ability to integrate diverse modalities makes them effective for generating more realistic and context-aware scenarios. We categoirze the usage of MLLMs in two categories further displayed in Table VI.

1) Safety-critical Scenario Generation: MLLMs enable richer scenario generation by leveraging diverse modalities such as video, GPS, crash reports, and potentially LiDAR point clouds, allowing for more realistic, structured, and risk-aware synthesis beyond the limits of image-based VLMs.

AutoScenario [177] presents a pipeline to generate realistic corner cases using multimodal crash data from NHTSA, including text, images, videos, and semi-structured reports. They use GPT-4o with CoT to generate structured scenario descriptions, which are then used to produce road networks in SUMO and agent behaviors in CARLA. Their scenario refinement is guided by GPS traces and frame-level similarity

TABLE VI
SUMMARY OF SCENARIO-GENERATION STUDIES USING MULTIMODAL LARGE LANGUAGE MODELS.

Category	Input					Model	Technique	Simulator	Output[1]	Paper
	Text	Image	Video	Dataset	Database					
Safety-critical Scenario (Section V-B1)	✓	✓	Real FPV	SUMO CARLA OSM	NHTSA GPS	GPT-4o	CoT	CARLA	🖹	AutoScenario [177]
ADAS Test Scenario (Section V-B2)	✓		Real FPV	HDD [178]		GPT-4V	CoT ICL	LGSVL	🖹	LEADE [179]

[1] Output icons: 〰 image, ⌒ Trajectory, 🖹 Scenario script.

between simulated and real scenes, to ensure good matching with the original crash event.

A promising future direction is to incorporate spatial modalities, such as LiDAR point clouds or BEV maps, to obtain a more accurate scene geometry and agent's localization, thereby improving realism beyond what 2D video and depth sensing alone can provide. This would better exploit the MLLMs' capabilities.

Metrics: (I) *Realism:* accuracy of the generated scene type, vehicles count, and object attributes compared to crash report descriptions; (II) *Diversity:* variation in structural elements such as lane count, agent density, and route length; (III) *Controllability:* success rate of generating executable and constraint-compliant scenarios, which was evaluated for instance in SUMO and CARLA.

2) ADAS Test Scenario Generation: The generation of test scenarios for ADAS aim to validate autonomous functions under typical but diverse conditions. MLLMs enable richer synthesis by reasoning over multimodal inputs, beyond the static perception limits of VLMs.

LEADE [179] generates ADAS test scenarios from real-traffic videos in the HDD dataset [178]. Key frames are used in multimodal ICL and CoT prompting with GPT-4V to create abstract scenarios, which are converted into executable programs for the LGSVL simulator [88]. The Apollo ADAS stack [180] runs an ego vehicle, and a dual-layer search identifies semantic-equivalent scenarios that expose behavioral differences between Apollo and human drivers.

Future work could align scenario generation with ADAS test standards, enabling the synthesis of regulation-compliant scenarios. Incorporating traffic rules and structured priors would also improve controllability and test coverage.

Metrics: The evaluation of MLLM-generated ADAS test scenarios in LEADE [179] focuses on the following. (I) *Accuracy:* element-wise extraction accuracy across categories like behavior, location, and object types from the video input; (II) *Correctness:* success rate of concrete scenario execution matching abstract semantics across different road types; (III) *Violation discovery:* quantifies the ability of the generated scenarios to expose safety-critical differences between the ADAS behavior and human driver behavior, including the number and diversity of detected violations.

C. MLLM-based Scenario Analysis

We list here the papers using MLLMs for scenario analysis in AD. We categorize the existing works into three key tasks:

1) Visual Question Answering (VQA): In comparison with VQA-based scenario analysis with VLMs, MLLMs have extended capabilities to deal with multi-modal sensor data such as videos, LiDAR point clouds, and HD maps, besides images and text. Based on their task and data modality, existing VQA datasets can be grouped into four categories:

(I) *General AD Tasks – Perception, Reasoning, and Control:* Several datasets target core autonomous driving tasks, including visual perception, reasoning, and decision-making. DriveGPT4 [169] introduces the first driving-specific, video QA-style instruction-following dataset, generated using GPT-4 with structured inputs including object detection bounding boxes, captions, and control signals formatted as text. It fine-tunes a multimodal LLM combining CLIP, a Valley tokenizer, and LLaMA2 with LoRA adapters to produce both textual explanations and control outputs. A mix-finetuning strategy merges general visual instruction data with driving-specific samples to improve reasoning and performance. VLAAD [170] introduces a multi-modal assistant for autonomous driving, trained on an instruction-following dataset derived from BDD-X and HDD videos, with QA pairs augmented using GPT-4. The model is built on Video-LLaMA, which combines a BLIP-2-based visual encoder, a Video Q-Former for temporal modeling, and a frozen LLaMA-2-7B language decoder. PEFT-MA is applied only to the Q-Former and projection layers, enabling the model to efficiently perform tasks such as VQA, free-form QA, ego-intention prediction, and scenario-level reasoning. LingoQA [127] presents a VQA dataset for autonomous driving, covering action, perception, and reasoning. It includes an Action set annotated with GPT-3.5 and a Scenery set generated by GPT-4 using CoT. The baseline model processes 5 video frames using CLIP and a Q-Former, with a linear projector to align features to Vicuna-1.5-7B's token space. PEFT-MA is applied to the Q-Former and projector; the LLM remains frozen. Evaluation is conducted using the novel Lingo-Judge classifier, which is trained with LoRA.

(II) *Spatio-Temporal Reasoning:* Datasets in this group emphasize reasoning over agent motion, temporal dependencies, and event semantics in driving scenarios. LiDAR-LLM [168] first tackles 3D captioning, grounding, and VQA from LiDAR point clouds. It extracts BEV features via a voxel encoder, embeds them using a View-Aware Transformer with learnable queries, which acts as a prior tokenizer, and projects them into the language space through an MLP. Adapter layers are fine-tuned within the LLM to

TABLE VII
SUMMARY OF SCENARIO-ANALYSIS STUDIES USING MULTIMODAL LARGE LANGUAGE MODELS.

Category	Input						Model			Technique[3]	Focus[4]	Paper
	Image	Context	Lidar	Video[1]	Map	Dataset	MLLM	LLM	Role[2]			
Visual Question Answering (VQA) (Section V-C1)		✓		FPV		BDD-X	CLIP +Valley +Llama2	GPT-4	MLLM: VideoQA Exec. LLM: QA Gen.	Projector LoRA	Perception Reasoning Control	DriveGPT4 [169]
	✓	✓		FPV		BDD-X HDD	BLIP2 + Llama2-7B	GPT-4	MLLM: VideoQA Exec. LLM: QA Aug.	QueryTrans Projector PEFT-MA	Prediction Reasoning	VLAAD [170]
	✓	✓		FPV	✓	In-house	CLIP + Vicuna1.5-7B	GPT-4	MLLM: VideoQA Exec. LLM: QA Gen.	QueryTrans Projector	Prediction Reasoning	LingoQA [127]
	✓	✓		Roads		TUMTraffic VideoQA	SigLIP + Qwen2-0.5B/7B	GPT-4omini	MLLM: VideoQA Exec. LLM: QA Gen.	MLP PEFT-MA Fft	ST Reasoning	TUMTraffic VideoQA [176]
	✓	✓		Multi View	✓	NuPlan	BEV Encoder Llama3.2V-11B	GPT-4o	MLLM: VideoQA Exec. LLM: MC-QA Gen.	MLP FusionTrans PEFT-MA	Perception ST Reasoning	NuPlanQA [165]
		✓		Multi View		nuScenes	Video-Llama		MLLM: VideoQA Exec.	Cross-attention ST-Adapter QueryTrans FusionTrans PEFT-MA	Perception Prediction Reasoning Risk	NuInstruct [166]
	✓	✓		Multi View		DRAMA [181]	ViT + MiniGPT-4	GPT-4o	MLLM: VideoQA Exec. LLM: VQA Aug.	ST-Adapter MLP Cross-Attention PEFT-MA	Perception Prediction Reasoning Risk	HiLM-D [171]
	✓	✓		FPV		SHRP2 [182]	Qwen2-VL-2B Qwen2-VL-7B 14 MLLMs	GPT-o1 Qwen2.5-72B	MLLM: VideoQA Exec. LLM: MC-QA Gen.	Fft ICL	Perception Reasoning Risk	DVBench [183]
		✓	✓			nuScenes	Voxel + Llama2-7B		MLLM: VQA Exec.	Encoder QueryTrans MLP PEFT-Adapter	Grounding Captioning	Lidar-llm [168]
	✓	✓	✓		✓	In-house	CLIP + Llama2-7B		MLLM: VQA Exec.	Projector LoRA	Perception	MAPLM [142]
Scene/Scenario Understanding (Section V-C2)	✓	✓				nuScenes	InternVl-1.5	GPT-4o	MLLM: Scene Under. LLM: QA Gen.	LoRA	Reasoning	InterDrive [174]
	✓	✓	✓			KITTI [184] nuScenes	Video-LLaVA GPT-4o			CoT	Reasoning	Jain et al. [185]
		✓		FPV		BDD-X	VideoMA +Ada-002 +OpenFlamingo			Cross-Attention LoRA	Reasoning	Dolphins [172]
	✓	✓	✓	Multi View		DriveLM	ViT-L/14 Llama-Adapter V			QueryTrans PEFT-Adapter PEFT-MA	Reasoning	Ishaq et al. [167]
		✓		FPV		BDD100K	ViT-L/14 +Vicuna-7B	GPT-3.5	MLLM: Video Capt. LLM: Caption Eval.	Projector QueryTrans PEFT-MA	Captioning	WTS [186]
		✓		FPV		LingoQA	LLaVA-VL-7B Qwen-VL-7B	Qwen2.5-1.5B Qwen2.5-7B	MLLM: Scene Extract. LLM: Scene Under.	Zero-shot	Captioning	V3LMA [187]
Risk Assessment (Section V-C3)	✓	✓			✓	DeepAccident [188]	GPT-4V	GPT-4	MLLM: Image Extract. LLM: Narrative Gen.	Zero-shot	Accident Prevention	AccidentGPT [189]
	✓	✓				DRAMA -ROLISP, DRAMA -SRIS [171]	ResNet-101 + Swin-L +Llama2-7B			QueryTrans Cross-Attention MLP PEFT-Adapter	Safety Interaction	MLLM-SUL [173]
	✓	✓		FPV		nuScenes	VideoLlama2	Llama3.1-8B	MLLM: Video Extract. LLM: Narrative Gen.	Zero-shot	Safety Interaction	ScVLM [190]
	✓	✓		FPV		DRAMA	Gemini1.5V-Pro			ICL	Risk-Reasoning	Abu et al. [191]

[1] Video: FPV = First Person View; BEV = Bird Eye Viewl;
[2] Role: Exec.= Execution; Aug. = Augmentation; Gen. = Generation; Under. = Understanding; Capt. = Captioning; Eval. = Evaluation; Extract. = Extraction;
[3] Techniques: Only focus on the techniques for MLLMs. Projector = Linear projector; MLP = MLP projection; QueryTrans = Query Transformer;
  FusionTrans = Fusion Transformer; PEFT-Adapter = Adapter layers; Fft = Full fine-tuning; PEFT-MA: Only trains modality alignment modules and LLMs are frozen;
  Encoder = Structure-aware encoder; CP = Contextual Prompting; ICL = In-Context Learning; CoT = Chain-of-Thought prompting;
[4] Focus: ST Reasoning = Spatio-temporal reasoning.

support 3D scene understanding. MAPLM [142] introduces a large-scale multimodal benchmark and VQA dataset and focuses on perception and HD map understanding in autonomous driving. It includes panoramic 2D images, BEV projections from LiDAR point clouds, and text descriptions extracted from HD maps. The baseline model aligns visual features using pretrained CLIP encoders and lightweight projection adapters, mapping them into the LLM's embedding space. Instruction tuning is performed via LoRA on Vicuna or LLaMA-2, enabling the model to perform effective scene-level reasoning across modalities.

(III) *Risk-Aware Reasoning:* To address safety-critical understanding, several datasets focus on risk recognition, intention estimation, and planning-related queries. TUMTraffic-VideoQA [176] introduces a multiple-choice video QA dataset targeting spatio-temporal reasoning in roadside traffic scenes, focusing on motion, object interaction, and event semantics. Visual metadata is extracted using standard detectors and captioned by off-the-shelf VLMs, while GPT-4o-mini generates QA pairs via template-augmented

prompting. The baseline model (TUMTraffic-Qwen) uses SigLIP for visual encoding, an MLP projector for modality alignment, and Qwen2 (0.5B/7B) as the LLM, which is fully fine-tuned for instruction-following QA. NuPlanQA [165] introduces a video QA dataset built on nuPlan, using GPT-4o to generate free-form QA pairs for training and multiple-choice QA for evaluation. To leverage this data, the authors propose BEV-LLM, an MLLM that integrates multi-view images and BEV features through a BEV encoder (BEVFormer), a BEV-Fusion module, and an MLP projector. The model uses LLaMA-3.2-Vision as a frozen backbone, while training is applied only to the BEV-Fusion module and projection layers, following a PEFT-MA strategy.

(IV) *Multi-Modal Extensions with LiDAR and HD Maps:* To extend reasoning beyond RGB data, some datasets incorporate 3D point clouds or HD maps. NuInstruct [166] introduces multi-view video QA datasets covering perception, prediction, risk, and planning tasks. QAs are generated via a structured SQL pipeline. The authors propose BEV-InMLLM, which extends MLLMs (e.g., Video-LLaMA) by utilizing ST-Adapters and a BEV-Injection module or a Fusion transformer that integrates spatial features from multi-view videos, resulting in improved performance on holistic autonomous driving tasks. HiLM-D [171] introduces DRAMA-ROLISP, a risk-aware VQA dataset enhanced via GPT-4. The model fine-tunes MiniGPT-4 with a ViT+ST-Adapter for video, a ResNet-based HRS encoder, and a P-Adapter for spatial fusion. A Query-Aware Detector integrates outputs for risk object localization and intention reasoning. The LLM itself remains frozen, with only the adapters, fusion, and projector layers fine-tuned. DVBench [183] introduces a comprehensive video-based VQA benchmark for safety-critical autonomous driving, built on SHRP2 [182] dashcam data. Multiple-choice QA pairs are generated and refined using GPT-4o and Qwen2.5-72B, covering two major categories—perception and reasoning—and 11 subcategories. The benchmark evaluates 14 MLLMs using the proposed GroupEval metric, which rotates answer positions to assess robustness. The authors also compare the performance of Qwen2-VL-2B/7B with and without full fine-tuning on the DVBench dataset.

A critical next step for VQA in autonomous driving is evaluating model robustness under out-of-distribution conditions. Existing datasets primarily contain frequent driving scenarios and well-structured questions, leaving models untested on rare events and objects, or unusual environmental conditions such as night, snow, or construction zones. Developing benchmarks that explicitly include such edge cases—and evaluating how well models generalize to them—is essential for deploying VQA systems in safety-critical, real-world driving environments.

Metrics: (I) *Answer Accuracy*, based on binary or multiple-choice correctness in response to task questions [142], [165], [166], [170], [176], [183]; (II)*Language Generation Quality*, measured using BLEU, CIDEr, METEOR, SPICE, and ROUGE-L scores to assess the fluency, relevance, and completeness of generated captions or free-form responses [127], [168], [171], [176]; (III) *Semantic and Spatial Grounding*, evaluated via metrics like mean Intersection-over-Union and L1/L2 distance errors to quantify grounding precision in risk detection and object localization tasks [171], [176]; (IV) *Robustness Evaluation*, performed using GroupEval [183], which tests model consistency by rotating the position of correct answers in multiple-choice settings.

2) Scene/Scenario Understanding: In autonomous driving, MLLMs play a crucial role in interpreting complex environments by integrating data from LiDAR, video, and HD maps. This subsection distinguishes between *Scene Understanding*, which focuses on static, image-based perception, and *Scenario Understanding*, which captures temporal dynamics, agent interactions, and evolving causal events. Finally, *Scenario captioning* emphasizes observable elements, reasoning targets spatial-temporal relations, intent inference, and causal analysis.

(I) Scene Understanding: InternDrive [174] and Jain et al. [185] focus on static scene understanding using image-based inputs. InternDrive proposes a framework for driving scenario understanding, covering perception, prediction, and reasoning, using MLLM. It generates QA pairs from nuScenes using GPT-4o, followed by human correction, and fine-tunes the MLLM InternVL-1.5 via LoRA on these annotations. The resulting model analyzes driving scenes from first-person view (FPV) images through visual instruction tuning. Jain et al. [185] evaluate MLLM for safety-critical scene understanding using QA pairs from KITTI and nuScenes across five categories. They benchmark Video-LLaVA and GPT-4V using merged image frames and textual LiDAR summaries, applying a CoT prompting approach to enhance multimodal reasoning without requiring true temporal modeling. Scenario understanding can be divided into two semantic levels:

(II) Scenario Understanding: In contrast, DOLPHINS [172], Ishaq et al. [167], WTS [186], and V3LMA [187] target scenario understanding, where temporal context, agent interaction, and causal reasoning are central. DOLPHINS [172] proposes an MLLM-based framework for human-like scenario and behavior understanding in autonomous driving. Built on OpenFlamingo, the model is instruction-tuned on image–instruction pairs using a Grounded Chain of Thought (GCoT) process for fine-grained reasoning. Then, it is adapted to driving videos using in-context examples retrieved via VideoMAE and Ada-002. Only the perceiver resampler, gated cross-attention, and LoRA modules are fine-tuned, enabling efficient alignment for multi-task driving understanding WTS [186] uses GPT-3.5 externally to generate human-guided ground truth captions and evaluate model outputs via LLMScore, which assesses semantic and syntactic similarity. The proposed Instance-VideoLLM combines CLIP ViT-L/14, a Video Q-Former, and Vicuna-7B, with fine-tuning applied to the adapter and Q-Former. The model is trained on enhanced video inputs incorporating bounding boxes, gaze data, and scene context, and is compared against other off-the-shelf MLLMs. Evaluating

performance on raw video inputs remains an open direction for future work.

(III) Scenario Captioning: This tasks reflects an intermediate level of understanding of driving scenarios. Ishaq et al. [167] propose a scenario-level spatial understanding framework that integrates short video clips, driving trajectories as text, and textual queries. They use a trajectory encoder and a Query Former to fuse the modalities, which are then passed into a frozen LLaMA-2 model with adapter layers. The model is fine-tuned by training both the Query Former and the adapters for efficient multimodal reasoning. V3LMA [187] proposes a fusion method that combines pre-trained LLMs and VLMs to enhance zero-shot 3D scenario understanding. They use off-the-shelf tools for grounding, object detection, and depth estimation to generate structured scenario descriptions, which are fed into the LLM. Visual features from an MLLM are then fused at either the feature level or the classification head. Despite being zero-shot, the model achieves competitive performance, comparable to fine-tuned MLLMs.

Current MLLMs for scene and scenario understanding primarily focus on short-term temporal contexts and curated question-answering tasks, which limits their capacity for deep reasoning. To advance toward high-level scenario reasoning, future work should explore long-range temporal modeling, causal inference across event sequences, and robust handling of out-of-distribution scenarios.

Metrics: (I) *Answer Accuracy*, which assesses correctness in structured tasks such as perception, planning, and reasoning [167], [174], [185]; (II) *Human Evaluation*, involving expert judgment of completeness, intent interpretation, and reasoning quality [172], [185], [186]; (III) *Spatial Reasoning Metrics*, which measure spatial understanding through localization accuracy [167].

3) Risk Assessment: The goals for the MLMMs include risk detection and violation inference for anticipating hazards, responsibility inference and scene-level safety scoring for analyzing incidents, and risk scoring and actionable advice generation for effective driver or system feedback. Together, these capabilities support comprehensive evaluation and response to dynamic traffic situations.

One approach to risk assessment emphasizes proactive hazard mitigation through interpretable scenario understanding. For example, AccidentGPT [189] combines multi-modal perception, such as images, 3D detections, BEV features, and trajectories, with GPT-4V for zero-shot scenario understanding dataset DeepAccident [188] and GPT-4 for safety evaluation using CoT and CP. It supports real-time accident prevention, post-accident analysis, and interactive safety decision-making through interpretable reasoning.

Other works focus on enabling interactive safety perception and feedback by aligning fine-grained visual inputs with language-based models. MLLM-SUL [173] fuses multi-scale visual inputs using ResNet-101 and Swin-L for low- and high-resolution features, combined via Query Formers and Gate-Attention based on the dataset Drama-ROLISP and Drama-SRIS from HiLM-D [171]. It fine-tunes LLaMA2-7B with adapters and applies an MLP head for scene captioning

and risk object localization. Similarly, ScVLM [190] proposes a multi-stage MLLM framework for risk assessment based on the nuScenes dataset, combining event type classification, conflict type identification, and narrative generation. It uses VideoLLaMA2 for zero-shot visual context extraction and LLaMA 3.1 8B to generate detailed descriptions of safety-critical events based on FPV driving videos.

A third direction emphasizes risk reasoning through structured question answering. Abu et al. [191] present a MLLM-based framework for safety-critical event detection using FPV videos from the DRAMA dataset. They compare Gemini-Pro-V1.5, Gemini-Pro-Video, and LLaVA using QA-based risk analysis with in-context learning, leveraging sliding window capture and textual context prompts to enhance event understanding.

Currently, we are missing the incorporation of richer contextual signals, such as driver's behavior, weather conditions, and map priors, for more holistic risk modeling. However, to ensure practical impact, it is critical to establish the reliability and determinism of LLM-based risk assessments. This remains a key challenge, as LLMs' behavior is inherently stochastic and may produce inconsistent outputs.

Metrics: (I) *Risk Localization Metrics*, such as mIoU and detection accuracy, are used to assess the precision of risk object identification [173]; (II) *Classification Metrics*, such as accuracy, mean Average Precision, and Area Under the Curve, are employed to evaluate the model's performance in categorizing both event types and conflict types [190].

VI. DIFFUSION MODELS (DMS)

This section provides an overview of DMs, explaining their underlying generative process and tracing their conceptual evolution. Given their generative nature, diffusion models excel at generating novel scenarios rather than analyzing existing ones. Accordingly, we survey their applications in scenario generation for AD, encompassing traffic flow synthesis, road layout design, image generation, and video generation. Finally, we discuss the evaluation metrics commonly used to assess the realism and quality of the generated scenarios.
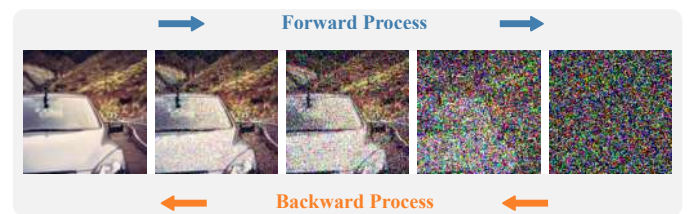

Fig. 7. An illustration of how a DM transforms a clean image into noise through the forward process, and then reconstructs it in reverse during the backward process.

A. Development of DMs

Diffusion models are generative models inspired by non-equilibrium thermodynamics, mirroring natural processes like ink diffusing through water or heat spreading in a metal

bar. At their core, they follow the simple yet powerful idea of systematically and gradually destroying structure in data through iterative noise addition and learning to reverse this process in a step-wise fashion [192]. While introduced by Sohl-Dickstein et al. [192], the approach gained widespread adoption through Ho et al.'s Denoising Diffusion Probabilistic Models (DDPM) [29]. The framework of DMs, as illustrated in Figure 7, involves two key phases: the forward process and the backward process.

1) Forward Process: The forward process refers to the act of gradually corrupting the original data x_0 by adding Gaussian noise over T steps, resulting in a sequence of noisy samples $x_1, x_2, \ldots, x_T$. By the final step, x_T, the sample is indistinguishable from pure noise.

2) Backward Process: To generate realistic samples from pure Gaussian noise, a DM must learn to invert its forward corrupting process. This is achieved through an iterative denoising procedure, where the model progressively refines the noisy input to recover the underlying data distribution. At each step, the model estimates and removes the noise added during the forward process, gradually reconstructing the target sample. Denoising is typically parameterized by a neural network, such as a U-Net [29], which is trained to accurately predict the noise component at each iteration, thereby enabling the synthesis of high-fidelity and diverse samples.

The basic diffusion pipeline functions as a weight-tied recurrent loop that iteratively transforms pure noise x_T into a clean sample x_0. Following the establishment of this paradigm, research advanced primarily along two key directions:

Controllability: Unlike the original DDPM, which is trained unconditionally and offers limited control over the output, a significant amount of research has introduced mechanisms to steer the diffusion process, making the output controllable. Conditioning the network on auxiliary signals, such as class labels, text embeddings, layout maps, or other modalities, enables structural constraints that guide the generative process. Classifier guidance [193] uses gradients from a separate classifier to steer sampling towards desired outputs. Classifier-free guidance [194] eliminates the need for a separate classifier by jointly training the model with and without conditioning signals, allowing adjustable control at inference. ControlNet [195] further expands controllability by incorporating spatial conditions such as edges, depth, or poses, enabling fine-grained user control.

Efficiency: The high computational cost of DDPM stems from many iterative steps at full resolution. Latent Diffusion Models (LDMs) [31] address this by operating in compressed latent spaces, reducing complexity while preserving quality. Diffusion Transformers (DiT) [196] build on this by replacing the U-Net with a transformer backbone, improving scalability and global context modeling.

These previous innovations have enabled the use of DMs across a wide range of domains. AD is a particularly impactful area where DMs are used to generate realistic scenarios efficiently and controllably.

B. Scenario Generation

This section deals with DMs for the generation of scenarios in AD. Table VIII displays the related literature papers, and both the table and this section are structured into four parts.

1) Traffic Flow Generation: Traditional simulators [50], [51], [88], [197] typically rely on replaying driving logs or using heuristic-based controllers, which often do not accurately capture the complexity and adaptability of real human behavior. Recent advancements in generative models present an opportunity to create realistic and diverse traffic behavior of virtual agents directly. These models can generate the traffic flow and therefore the behavior (trajectories) of multiple agents over time. To have impact for the AD development, it must achieve both realism and controllability, ensuring simulations reflect human-like driving behaviour while adhering to customizable rules. However, the mechanisms for achieving controllability differ:

(I) *Gradient-Based Guidance* in DMs works by modifying the predicted mean at each denoising step using the gradient of a control objective. This perturbs the generation toward samples that better fulfill the objective while still following the underlying diffusion process. CTG [198] incorporates Signal Temporal Logic (STL) to encode traffic rules, using the robustness score of STL as a measure of how well the rules are followed and leveraging its gradient to guide trajectory sampling. CCDiff [206] leverages the gradient of a constrained Markov decision process (MDP) to guide trajectory generation for multiple agents, with the MDP encoding specific control goals such as causing collisions. Before applying guidance, a causal reasoner ranks agents based on inter-agent influence and restricts guidance to the most impactful subset to improve efficiency and effectiveness. DiffScene [207] defines three differentiable objectives: safety-critical (maximizing collision risk), functional (hindering ego task completion), and constraint-based (enforcing realism rules). Lu et al. [208] extend DiffScene by encouraging adversarial agents to exhibit aggressive maneuvers (via acceleration/yaw rate variability) and manipulate traffic density around the ego vehicle. AdvDiffuser [209] trains a model to predict how likely a scenario causes failures for a given AV planner and uses this signal to guide the sampling process. SafeSim [210] and VBD [211] generate potential trajectories and identify those that would lead to collisions, then use guided diffusion to denoise them. A different approach is proposed by Zhong et al. [212] and Peng et al. [213] (LD-Scene), both of which leverage an LLM to translate natural language instructions (e.g., "aggressive lane change") into differentiable guidance functions, bridging high-level intent with low-level control.

(II) *Architecture Conditioning* embeds the control signal directly within the network's structure so that constraints are enforced throughout each iteration, rather than being injected afterwards as an external correction. DM achieve this by accepting extra conditioning inputs, such as tokens that carry agent attributes, scene statistics, language descriptions, or spatial masks. These additional inputs are processed by dedicated layers, for example, cross-attention blocks or inpainting modules, and are fused with the latent scene

TABLE VIII
SUMMARY OF SCENARIO GENERATION STUDIES USING DIFFUSION MODELS.

Category (Output)	Safety critical scenario?	Input				Controll-ability[1]	Controllable Factor[2]	Technique	Base Model	Dataset	Paper
		Road Topology	Initial State	Text Prompt	Bounding Boxes						
Traffic Flow (Sec VI-B1)	No	✓	✓			◐	Speed Goal Waypoint	STL as Guidance	DDPM	nuScenes	CTG [198]
				✓		●		LLM-Driven Scene Initialization	DiT	Argoverse 2	DriveGen [199]
		✓				◐	Traffic Density Agents' Position Agents' Speed Agents' Size	Architecture Conditioning	LDM	Argoverse 2	Pronovost et al. [200]
		✓				◐	Traffic Density	Architecture Conditioning	DiT	WOMD	SceneDiffuser [201]
						○		Map-Free Scene Generation	LDM	WOMD	DriveSceneGen [202]
						○		Raster-to-Vector Representation	DiT	nuPlan	Sledge [203]
				✓		◐	Traffic Density Road Layout	Vectorized Latent Diffusion	LDM	WOMD nuPlan	Rowe et al. [204]
		✓	✓			◐	Speed Goal Waypoint	Preference Optimization	DiT	nuScenes	Yu et al. [205]
	Yes	✓	✓			◐	Collision Type	MDP as Guidance	DDPM	nuScenes	CCDiff [206]
		✓	✓			◐	Speed	Gradient-Based Guidance	DDPM	CARLA	DiffScene [207]
		✓	✓			◐	Traffic Density Speed	Gradient-Based Guidance	DDPM	nuScenes	Lu et al. [208]
		✓	✓			○		Gradient-Based Guidance	LDM	nuScenes	AdvDiffuser [209]
		✓	✓			◐	Dirving Style Collision Type	Partial Diffusion	DDPM	nuPlan nuScenes	SafeSim [210]
		✓	✓			◐	Driving Style	Gradient-Based Guidance	DiT	WOMD	VBD [211]
		✓	✓	✓		●		LLM-Generated Loss Function	DiT	nuScenes	Zhong et al. [212]
				✓		●		LLM-Driven Scene Initialization	LDM	nuScenes	LD-Scene [213]
Static Traffic Element (Sec VI-B2)	No			✓		◐	Number of Lanes Type of Road	Road-UNet architecture	DDPM	OSM	DiffRoad [214]
		✓				○		End-to-End Differentiable	LDM	In-house	Pronovost et al. [215]
		✓				◐	Agents' Position Agents' Density Agent' Speed Agents' Size	Guided Agent Placement	DDPM	Argoverse 2	SceneControl [216]
Driving Image (Sec VI-B3)	No			✓		◐	Road Topology Traffic Density Weather	Structured Prompt	LDM DDPM	nuScenes	Text2Street [217]
					✓	◐	Camera Pose	Bounding Box Translation	LDM	nuSences	GeoDiffusion [218]
			BEV Sketch			◐	Weather Lighting Condition	Controller & Coordinator	LDM	nuScenes	BEVControl [219]
		✓		✓		◐	Camera Pose Weather Lighting Condition	Cross-View Attention	LDM	nuScenes	MagicDrive [220]
		✓		✓	✓	◐	Weather Lighting Condition Camera Pose	Dual-Branch Diffusion	LDM	nuScenes	DualDiff [221]
Driving Video (Sec VI-B4)	No		BEV Sequence			◐	Weather Lighting Condition Landscape	4D Attention	LDM	nuScenes	Panacea [222]
			3D Layout Sequence			◐	Weather Lighting Condition	Cascaded Video Synthesis	LDM	nuScenes	DrivingDiffusion [223]
		✓		✓		●		Multi-Control Distillation	DiT	nuScenes	DiVE [224]
			Canny Edge Map Depth Map Text Prompt			◐	Weather Lighting Condition	Dual-Branch Diffusion	LDM	DriveScene-DDM [225]	DcTDM [225]
			Initial Frames			○		Frame Sampling Scheme	DDPM	WOMD	DriveGenVLM [226]
		✓		✓	✓	◐	Weather Lighting Condition	Dual-Branch Diffusion	LDM	nuScenes Waymo Open	DualDiff+ [227]
				✓		◐	Weather Traffic Density Landscape	Adapting Existing Methods	LDM	KITTI	GenDDS [228]
	Yes			✓		●		Temporal Shift Adapter	LDM	DoTA [229]	DrivingGen [230]
				✓		●		Adapting Existing Methods	DiT	MM-AU [231]	AVD2 [232]

[1] Controllability: ● Full control (users can fully customize scenes); ◐ Partial control (supports specific parameter adjustments); ○ No control.
[2] Only models with partial controllability are discussed here in this column. Fully controllable models can follow any input (typically via LLMs), and models without control fall outside the scope of this discussion.

representation at each denoising iteration. Pronovost et al. [200] encode agent attributes (speed, heading) and global scene properties (agent density) as tokens processed by cross-attention layers. SceneDiffuser [201] frames trajectory generation as an inpainting task on a 3D tensor of shape $A \times T \times D$, each representing agents, timesteps, and features. Scene editing and agent injection are made possible by adjusting the scene tensor and the associated inpainting mask. DriveGen [199] uses a natural language description to generate road layouts and place vehicles via an LLM.

A VLM is applied afterwards to analyze the BEV to identify potential future goals. Finally, a DM generates realistic trajectories from each vehicle's initial state to its predicted goal. DriveSceneGen [202] addresses two key problems: scene initialization and rollout. It first synthesizes a BEV image of road layouts and agent positions using a DM, then vectorizes the output for trajectory prediction with a Motion Transformer (MTR). SLEDGE [203] and ScenarioDreamer [204] address the same task but optimize the generation pipeline. Specifically, SLEDGE introduces a

raster-to-vector autoencoder to compress scenes into latent maps for further diffusion, whereas ScenarioDreamer further advances this by operating the DM directly in vector space. Together, these methods reflect a progression from pixel-level (DriveSceneGen) to compressed-raster (SLEDGE) to fully vectorized (ScenarioDreamer) generation.

(III) *Preference Optimization* departs from both gradient-guidance and architecture-conditioning. Instead of incorporating explicit control signals or manually defined differentiable loss functions, Yu et al. [205] use Direct Preference Optimization (DPO) to fine-tune the DM. During this process, the model generates two candidate trajectories for a given scene and a rule set. These are scored using rule-based heuristics, and the model is updated to prefer the "better" trajectory, effectively learning control preferences without manually crafting individual loss terms.

Despite recent progress, diffusion-based traffic flow generators still depend on hand-designed control inputs. Gradient-guided models require carefully tuned objective weights, whereas architecture-conditioned models must fix in advance the token or mask schema that encodes each rule; extending them to new constraints demands costly retraining or extensive fine-tuning.

Metrics The evaluation of DMs for traffic flow generation mainly focuses on assessing the realism and controllability of the generated trajectories.

(I) *Realism* is evaluated by measuring how closely generated trajectories mirror natural human driving. The Wasserstein Distance (WD) is widely used [198], [207], [208], [210], [212] to quantify distributional discrepancies in motion dynamics, particularly in longitudinal/lateral accelerations and jerk. These distributional discrepancies are also measured using the Kullback-Leibler Divergence (KL) [207]. Path geometry is evaluated through metrics such as the Fréchet Distance [204], [207], which identifies maximum positional deviations between trajectories, and the Symmetric Segment-Path Distance (SSPD) or Dynamic Time Warping (DTW) [207], which compare spatial deviations and temporal alignment. Geometric realism is further enforced through off-road rate calculations [198], [200], [205], [206], [210], [211], [213], ensuring vehicles remain within drivable areas, and Lane Heading Difference (LHD) [200], which checks alignment between vehicle orientation and lane direction. Collision rate is utilized in a context-dependent way: models [206], [208], [209], [213] targeting safety-critical scenarios may intentionally generate higher collision rates to test edge cases, whereas those [198], [200], [204], [207], [210], [211] focused on normal driving minimize it to reflect real-world behavior. Collectively, these metrics ensure that generated trajectories align with statistical properties of real data, adhere to geometric constraints, and satisfy scenario-specific safety objectives.

(II) *Controllability* lacks standardized evaluation due to varying definitions of "control signals" across studies. Most works focus on goal-directed metrics, such as adherence to speed limits [198], [205], [207], [208], [210], [212] or reaching predefined waypoints [198], [205], [212]. A smaller subset [198], [212] also evaluates rule-based controllability by

measuring how well the agents comply with traffic rules.

2) Static Traffic Element: Several diffusion-based models have also been developed to generate various autonomous driving components beyond agent trajectories.

DiffRoad [214], for instance, synthesizes 3D road layouts from structured text inputs (e.g., "two three-way intersections") and evaluates the outputs based on criteria like smoothness and semantic validity (e.g., detecting overlapping segments).

Pronovost et al. [215] and SceneControl [216] focus on generating the initial placement of the agents for downstream traffic simulation. Pronovost et al. introduce a scene autoencoder that compresses rasterized agent layouts into latent embeddings. A DM, conditioned on a road map, is then trained over these embeddings, and a decoder reconstructs oriented bounding boxes for the agents. SceneControl offers additional flexibility through guided sampling, allowing fine-grained user control (e.g., enforcing speed constraints) and realism guarantees (e.g., collision avoidance and lane adherence) during the generation process. To assess how well the generated scenes match real-world data, both methods compare statistical distributions between real and synthetic datasets.

These static-scene generators still have notable gaps. When DM is used to synthesize road layouts, fine-grained elements such as traffic signs, signals and lane markings are often omitted. As a result, the resulting maps lack the fidelity needed for high-realism driving simulation. Moreover, initial-scene generators are also highly map-specific: they absorb the spatial priors of the training corpus and can place agents unrealistically when applied to unseen road geometries or regions with different driving conventions.

3) Image Generation: Reliable perception in AD systems relies heavily on large-scale camera datasets with rich annotations. Rather than laboriously collecting and annotating street-view images from the natural environment, DMs offer a promising solution by generating realistic images for synthesizing street-view data.

Text2Street [217] decomposes structured prompts, such as "a street view image with a crossing, 4 lanes, 3 cars, 2 persons, and 2 trucks on a sunny day", into three distinct components: road topology, object layout, and weather condition. Each of these components is handled by a dedicated DM. The first model processes the road topology to generate a BEV road layout. The second model takes this BEV layout and incorporates the object layout, producing a map that includes vehicles, pedestrians, and other foreground elements. The third model transforms this BEV representation into a realistic camera-view street scene. To handle geometric conditions more effectively, GeoDiffusion [218] converts bounding boxes into textual prompts that guide a pre-trained text-to-image DM. This involves translating continuous bounding box locations into discrete tokens and developing methods to balance the visual prominence of foreground objects with the often-dominant background regions during image generation. Baresi et al. [233] generate rare OOD driving scenarios (e.g., snow, desert) using three diffusion-based strategies: instruction

editing, inpainting, and inpainting with refinement.

Meanwhile, other works have focused on generating multi-view images. BEVControl [219] addresses the complexity of editing dense segmentation maps by using editable BEV sketches as input. It introduces a "controller and coordinator" mechanism to ensure that generated objects match the sketch accurately and maintain consistency across multiple viewpoints. MagicDrive [220] considers road layouts, bounding boxes, camera poses, and textual descriptions such as weather and time of day as input. It introduces a cross-view attention module that allows each camera view to access information from its immediate neighbors, ensuring visual consistency and coherence across all generated views. DualDiff [221] employs a distinct approach through its dual-branch architecture, which separately handles the generation of foreground and background. The method begins by projecting 3D occupancy data onto camera planes to create dense feature maps. These projected features are fused with other inputs (3D bounding boxes for the foreground and vectorized road maps for the background). Each branch processes its respective features independently, and their outputs are ultimately combined to generate the final synthesized image.

Despite recent advances, fine details such as traffic signs, pole-mounted signals and lane markings are frequently simplified or omitted, resulting in generated images that fail to cover many visual corner cases that real perception stacks must handle. Photometric realism is also limited: simplified lighting models and the absence of camera artifacts such as rolling-shutter distortion, lens flare, and sensor noise create a noticeable domain gap when these synthetic frames are used to train or evaluate real-world detectors.

Metrics: (I) *Realism* is primarily evaluated using the Fréchet Inception Distance (FID) [217]–[221], which quantifies the statistical similarity between generated and real image distributions. (II) *Semantic Alignment* is assessed through the CLIP Score, derived from the Contrastive Language–Image Pretraining model [27]. This metric calculates the cosine similarity between image embeddings and text (or pairs of images) in a shared feature space, measuring how well visual content aligns with semantic intent. Applications vary by context: models like Text2Street [217] use text-to-image CLIP scores to validate adherence to textual prompts, while BEVControl [219] employs image-to-image CLIP scores to ensure visual consistency between adjacent camera views in multi-perspective systems. (III) *Control Capability* is mostly evaluated using a pre-trained object detector or segmentation model. The object detector assesses how well the generated foreground objects align with the control signals (e.g., generating specific objects such as vehicles or pedestrians) [218]–[221].

4) Video Generation: Recent work has also demonstrated significant progress in generating realistic driving videos using DMs, addressing challenges such as temporal consistency, controllability, and data diversity.

Several studies have introduced innovative architectures to ensure multi-view and temporal consistency in generated videos. Panacea [222] generates multi-view video sequences by first synthesizing images from BEV inputs and then expanding them along the temporal dimension. The method introduces a 4D attention mechanism that takes into account intra-view (within each camera), cross-view (between adjacent cameras) and cross-frame (between temporal patches). DrivingDiffusion [223] also employs a multi-stage approach: it first generates a consistent initial frame across all camera views from a layout, then uses a temporal model to produce short view-specific sequences, and finally refines long-term consistency via a sliding-window post-processing module. DiVE [224] focuses specifically on efficient multi-view driving scene generation. It introduces Multi-Control Auxiliary Branch Distillation (MAD) to streamline multi-condition classifier-free guidance, significantly reducing inference time. DiVE also proposes view-inflated attention, a lightweight mechanism enforcing cross-view consistency without adding parameters.

Another strategy for video generation is adapting image DMs with temporal expansion. DrivingGen [230] takes this approach by adapting a pre-trained text-to-image DM for video generation. To expand the T2I into the temporal dimension, it incorporates a temporal shift adapter that efficiently propagates information across frames using modified 2D convolutions, rather than expensive 3D operations. Similarly, DcTDM [225] leverages dense depth maps and canny edge maps as dual conditioning inputs to extend an image DM for video generation. DriveGenVLM [226] makes significant contributions to long-term video generation through its innovative conditioning strategies and sampling schemes, including frame-by-frame generation and keyframe interpolation approaches that offer distinct trade-offs between quality and speed.

In contrast, DualDiff+ [227] generates videos through a dual-branch architecture that decouples foreground and background modeling. The model first projects a 3D occupancy grid into 2D space and then fuses these features with semantic inputs, including 3D bounding boxes (foreground) and maps (background).

Another line of research focuses on practical implementations through model adaptation techniques. GenDDS [228] fine-tunes Stable Diffusion XL [234] using LoRA [94] to produce driving scene images, which are then extended into coherent videos through a temporal transformer in Hotshot-XL [235]. AVD2 [232] fine-tunes the Open-Sora 1.2 model [236] on the MM-AU [231] dataset to generate videos annotated with accident causes and avoidance strategies.

Despite recent advances, diffusion-based generators for driving videos still face significant challenges. They often struggle to maintain consistent temporal and multi-view coherence, particularly over extended clips. Additionally, their understanding of physical dynamics remains limited—vehicles may behave in ways that defy inertia or violate occlusion logic.

Metrics. *(I) Realism* is assessed using metrics like Fréchet Video Distance (FVD) [222]–[227], [230], [232], which compares the distribution of generated video clips to real ones using features from a spatiotemporal network, and frame-wise

Fréchet Inception Distance (FID) to evaluate individual frame quality [222]–[225], [227], [232]. Kernel Video Distance (KVD) addresses non-Gaussian feature distributions, offering a more flexible alternative to FVD [230]. *(II) Controllability* employs pretrained video-based or image-based object detectors to verify that required objects (e.g., vehicles, pedestrians) are properly generated [222]–[225], [227]. In addition, consistency across different viewpoints is captured by the View Matching Score (VMS) [222], which measures cross-view coherence in multi-view video generation, ensuring that scenes remain semantically and spatially consistent from varying perspectives.

VII. WORLD MODELS (WMs)

World Models are generative neural network models that learn compressed spatial and temporal representations of an environment [33]. They enable agents to develop an internal model of the world to make predictions about future states of the surrounding world environment, concerning both dynamic and static objects. In this section, we focus on their scenario generation capability, categorizing recent works into visual, 3D occupancy, and multi-modal generation. We also discuss related architectural innovations and benchmarks.

A. Development of World Models

1) Evolution of World Models: The development of WMs centers on learning compact and predictive representations of the environment's dynamics, enabling agents to reason about future events in a latent space. Specifically, WMs consist of an *encoder-decoder* paradigm [33], [236]–[238].
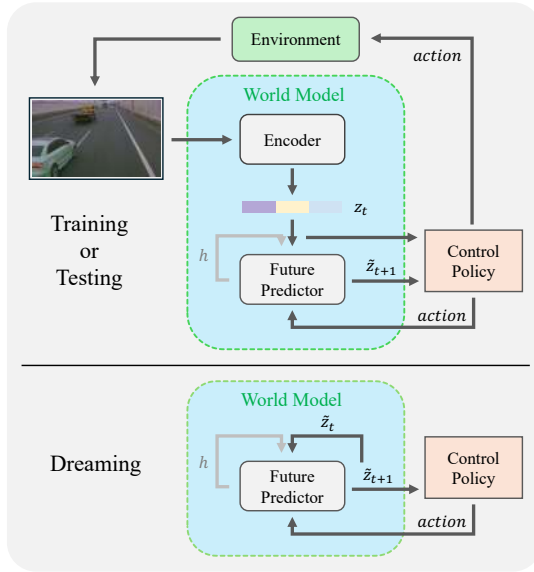

Fig. 8. An overview of the world model's training/testing and its dreaming process. For training/testing (above part), z_t is the latent representation of the input observation (e.g., image). $\tilde{z}_{t+1}$ is the prediction of the latent representation for the next time step. h is the hidden state that contains historical information. For the dreaming (bottom part), $\tilde{z}_t$ is the input latent vector of the future predictor, and it is usually initialized with a $\tilde{z}_0$ and later comes from the output of the future predictor from the last time step (auto-regressive model).

As illustrated in Figure 8, the *encoder* is used to encode the multimodal input (images, point clouds, 3D occupancy voxels, etc.) into a latent vector. Then, the future predictor (*decoder*) predicts the future latent representation based on the latent vector together with the action from the control policy's output. When the WM's pre-training is finished, the future predictor can not only be used as a motion prediction model in the testing phase, but can also be used as a tool for "dreaming" (i.e., generation) of new scenarios that likely do not exist in the original dataset. In this regard, WMs can be effectively utilized to generate data that falls outside the original dataset's data distribution. This capability is particularly valuable for autonomous driving, where rare but critical scenarios may be underrepresented in collected datasets, yet are crucial for testing the robustness of autonomous driving models.

2) Architectural Components of World Models: As introduced by Ha and Schmidhuber [33], WMs typically consist of two key components: a Vision model (V) and a Memory model (M). The vision model (called *encoder* in Figure 8), often implemented as a Variational Autoencoder (VAE), compresses high-dimensional observations into a compact latent representation. This dimensionality reduction creates a manageable state space for prediction and generation. The memory model (called *future predictor* in Figure 8) was a recurrent network (e.g., LSTM or GRU) in the early implementations. The memory model captures temporal dependencies and dynamics across sequential observations, enabling the prediction of future states. Modern WMs for AD have evolved this basic architecture to incorporate advanced techniques into the memory model (future predictor). For example, GAIA-1 [239] uses a transformer, and the newer GAIA-2 [241] employs a latent diffusion model (LDM) [31] for future prediction and generation. Very recently, Diffusion Transformers (DiT) [196], Stable Video Diffusion (SVD) [242] models and videoLDM [247] have gained popularity as core architectures for WMs.
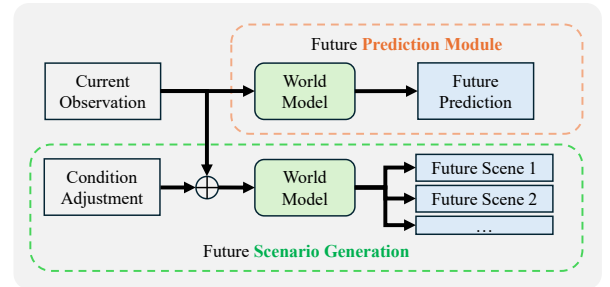
Fig. 9. An illustration of how research on WMs for AD can be broadly categorized into two main functions: future prediction of agents' motion, and future scenario generation.

WMs can generally be used for two purposes in AD: future motion prediction [273] and scenario generation [239], [241].

As illustrated in Figure 9, in this section we focus primarily on the application of WMs for scenario generation. The different papers, their corresponding dataset and the available code is displayed further in Table IX.

TABLE IX
COMPARISON OF KEY RESEARCH ABOUT WORLD MODELS FOR SCENARIO GENERATION IN AUTONOMOUS DRIVING.

Category	Input						Controllability[2]	Multi-view Generation	World Model Architecture	Model Types	Dataset	Code[3]	Paper
	Image	Text	Action	Trajectory	Geometry[1]	Map							
Visual Generation (VII-B1)	✓	✓	✓				◐		Autoregression	Transformer [21]	In-house		GAIA-1 [239]
	✓	✓	✓		✓	✓	●	✓	Diffusion	LDM [31]	nuScenes [121]	✓	DriveDreamer [34]
	✓	✓					○		Diffusion	LDM	nuScenes, In-house		ADriver-I [240]
	✓	✓	✓	✓	✓		●	✓	Diffusion	LDM	In-house		GAIA-2 [241]
	✓	✓	✓	✓	✓	✓	●	✓	Diffusion	SVD [242]	nuScenes	✓	DriveDreamer-2 [35]
	✓	✓			✓	✓	◐		Diffusion	LDM	Waymo Open dataset [72]	✓	DriveDreamer4D [243]
	✓	✓	✓	✓			●		Diffusion	SVD	nuScenes, etc [72], [146], [244].	✓	Vista [245]
	✓			✓			○		Autoregression	Transformer	nuPlan [48], In-house	✓	DrivingWorld [246]
	✓	✓	✓		✓	✓	●	✓	Diffusion	VideoLDM [247]	nuScenes		Drive-WM [248]
	✓	✓			✓	✓	◐	✓	Diffusion	LDM	nuScenes	✓	MagicDrive [220]
		✓	✓	✓	✓	✓	◐	✓	Diffusion	LDM	nuScenes		MagicDrive3D [249]
	✓	✓	✓	✓	✓	✓	●	✓	Diffusion	DiT [196]	nuScenes		MagicDrive-V2 [250]
		✓		✓	✓	✓	◐	✓	Diffusion	LDM	nuScenes, Occ3d [251]	✓	WoVoGen [252]
	✓			✓	✓	✓	◐		Diffusion	SVD	Waymo Open dataset	✓	ReconDreamer [253]
	✓	✓		✓	✓	✓	◐	✓	Autoregression	LDM	nuScenes		DualDiff+ [227]
	✓	✓		✓	✓	✓	◐	✓	Diffusion	LDM	nuScenes		Panacea [222]
	✓	✓		✓	✓		◐		Diffusion	DiT [196]	Cosmos [254]		Cosmos-Transfer1 [255]
	✓			✓	✓		◐		Diffusion	VideoLDM [247]	nuScenes		GeoDrive [256]
3D Occupancy Generation (VII-B2)				✓			○		Diffusion	DiT [196]	nuScenes	✓	OccSora [257]
	✓	✓	✓	✓			●		Autoregression	Transformer	nuScenes, Lyft-Level5 [258]	✓	Drive-OccWorld [259]
				✓	✓		○		Diffusion	Latent DiT [260]	nuScenes	✓	DOME [261]
					✓		○		Autoregression	Transformer	nuScenes		RenderWorld [262]
		✓		✓	✓		◐		Autoregression	Transformer	nuScenes, etc [128], [251].		OccLLama [263]
	✓	✓	✓				◐		Autoregression	Transformer	nuScenes, Openscene [264]		DriveWorld [265]
Multi-modal Generation (VII-B3)	✓	✓			✓	✓	◐	✓	Autoregression	Transformer	nuScenes		HoloDrive [266]
	✓		✓		✓		◐	✓	Diffusion	DDIM [30]	nuScenes, Carla		BEVWorld [267]
	✓			✓	✓		◐		Diffusion	SVD [242]	BDD [268], etc [244], [269].	✓	GEM [270]
Evaluation & Benchmark (VII-B4)	✓	✓	✓	✓			●		Autoregression	Transformer	nuScenes, In-house	✓	ACT-Bench [271]
	✓	✓		✓	✓	✓	◐	✓	Diffusion	SVD	nuScenes	✓	DriveArena [272]

[1] Geometry means 3D geometric representation and includes: 3D voxel occupancy, 3D bounding box, 3D depth, 3D segmentation and 3D point cloud.
[2] Controllability: ● Full control (models offer fine-grained scene customization with flexible control over scene elements); ◐ Partial control (models support limited or parameterized control (e.g., adjusting map); ○ No control.
[3] Code availability: "-" means code is either not announced to be released in the paper, or not fully released yet; "✓" means code is released open-source.

B. Scenario Generation with World Model Dreaming

World model dreaming [33] refers to the process of using a trained world model to generate new scenarios by sampling from its learned latent space without additional real-world inputs. Once a world model has captured the underlying dynamics of an environment, it can "dream" new scenarios that follow similar physical and logical patterns as those in the training data, but with novel combinations of elements and conditions that may not have been observed during training. As shown in Table IX, recent research on WMs for AD can be categorized into four groups:

1) Visual Generation: Visual generation approaches focus on creating realistic driving scenarios through the synthesis of images and videos. They represent the most mature category of WM applications in AD. GAIA-1 [239] pioneered the use of generative world models for AD, by demonstrating the ability to generate diverse traffic scenarios with multiple interacting agents. GAIA-1 casts world modeling as an unsupervised sequence modeling problem, mapping multimodal inputs (video, text, and action) to discrete tokens and predicting subsequent tokens. This approach enables fine-grained control over ego-vehicle behavior and scene features, exhibiting emerging properties such as contextual awareness and 3D geometry understanding.

GAIA-2 [241] significantly advances the GAIA-1 paradigm through a latent diffusion WM that supports controllable video generation conditioned on structured inputs (e.g., ego-vehicle dynamics and agent configurations). GAIA-2 generates high-resolution, spatio-temporally consistent multi-camera videos across diverse driving environments and countries (UK, US, Germany), making it a versatile tool for complex scenario simulation with good multi-view consistency and control precision.

DriveDreamer [34] addresses the limitations of prior WMs by developing a model entirely derived from real-world driving scenarios. Using its Autonomous Driving Diffusion Model (Auto-DM) and a two-stage training pipeline, DriveDreamer first learns traffic structural constraints and then anticipates future states through video prediction. This approach excels in generating controllable driving videos and predicting driving policies, thereby enhancing perception tasks such as 3D detection.

DriveDreamer-2 [35] extends the DriveDreamer framework [34] by incorporating a large language model to generate user-defined driving videos. DriveDreamer-2 converts user queries into agent trajectories, generates High-Definition (HD) Maps that comply with traffic regulations, and employs a

unified multi-view model to ensure temporal and spatial coherence. It can also produce uncommon scenarios, such as abrupt vehicle cut-ins, in a user-friendly manner.

DriveDreamer4D [243] extends the DriveDreamer framework to 4D (spatio-temporal) scene representation. By modeling dynamic scenes in four dimensions and incorporating map, layout, and text conditioning, it enhances the realism and utility of the generated data to train perception and planning algorithms.

ADriver-I [240] introduces a unified WM that differs from traditional modular designs by using interleaved vision-action pairs to standardize visual features and control signals. Leveraging multimodal large language models (MLLM) and diffusion techniques, it autoregressively predicts control signals and forecasts future frames, creating a continuous simulation loop.

DrivingWorld [246] introduces a GPT-style world model for autonomous driving, featuring spatial-temporal fusion mechanisms. It employs next-state and next-token prediction strategies to model temporal coherence and spatial information, implementing masking and reweighting strategies to mitigate long-term drifting and improve 3D detection and motion forecasting.

MagicDrive [220] is a framework for street view generation with diverse 3D geometry controls, including camera poses, road maps, and 3D bounding boxes, along with textual descriptions. It addresses the challenge of 3D control in traditional diffusion models, offering high-fidelity image and video synthesis with nuanced 3D geometry and cross-view attention for multi-camera consistency.

MagicDrive3D [249] presents a pioneering pipeline for controllable 3D street scene generation that supports multi-condition control, including BEV maps, 3D objects, and text descriptions. Unlike methods that reconstruct scenes before training, it first trains a video generation model and then reconstructs 3D scenes from generated data, enabling flexible and high-quality scene reconstruction for any-view rendering.

WoVoGen [252] introduces a world volume-aware diffusion model for generating controllable multi-camera driving scenes. It operates by predicting explicit 3D world volumes to guide video generation, ensuring that multi-camera perspectives align accurately with the underlying scene geometry, and maintaining high spatial and inter-sensor consistency.

ReconDreamer [253] focuses on crafting world models for driving scene reconstruction through online restoration. It emphasizes online learning for real-time applications, allowing continuous updates to the world model as new data is acquired, which is critical for adaptability to changing conditions in autonomous driving.

DualDiff+ [227] introduces a dual-branch diffusion model for high-fidelity video generation. The model incorporates Occupancy Ray Sampling (ORS) for semantic-rich 3D representation and Semantic Fusion Attention (SFA) to integrate multi-modal data with a foreground-aware masked (FGM) loss, thereby improving the generation of small objects.

GeoDrive [256] integrates 3D geometry conditions into driving world models. It enhances spatial understanding and action controllability through 3D representation extraction, 3D video rendering with dynamic editing, and dual-branch control for spatio-temporal consistency, achieving exceptional video quality with minimal training data.

Metrics: Similarly to scenario generation with diffusion models (Sec. VI), frame-wise *Fréchet Inception Distance* [274] and *Fréchet Video Distance* [275] are used to evaluate the quality of WMs' visual generation.

2) 3D Occupancy Generation: 3D occupancy generation predicts and synthesizes volumetric representations of driving environments, capturing both the spatial structure and the temporal dynamics of scenes.

OccSora [257] presents a novel framework to predict 4D occupancy scene evolution in AD, treating it as a video prediction task. OccSora employs a 4D scene tokenizer to obtain compact spatio-temporal representations. Then, it trains a diffusion transformer to generate 4D occupancy conditioned on trajectory prompts, enabling trajectory-aware simulation of various driving scenarios.

Drive-OccWorld [259] combines a planner with a dynamic WM to predict 3D occupancy and flow from multi-view images. It uses motion-aware BEV sequences as an intermediate representation, integrating multi-view video data with motion cues to achieve robust predictions of both static and dynamic elements in complex urban environments.

DOME [261] performs 3D occupancy prediction using a continuous variational autoencoder-like tokenizer to preserve intricate spatial information. Unlike discrete tokenization methods, DOME's continuous approach captures subtle geometric details while maintaining computational efficiency, employing probabilistic modeling to enhance robustness to sensor noise and occlusions.

RenderWorld [262] focuses on fine-grained occupancy prediction through a novel tokenization strategy. This approach captures spatial relationships, improving prediction accuracy for both static and dynamic objects while balancing granularity and computational efficiency.

DriveWorld [265] focuses on 4D pre-trained scene understanding, separating static spatial context from dynamic temporal changes to enable precise occupancy prediction from multi-view videos. The model relies on self-supervised learning to reduce reliance on annotated data, thereby enhancing scalability and improving downstream tasks such as perception and planning.

Even though some of these recent 3D occupancy WMs research have shown good capability of predicting the evolution of driving environments in a volumetric way, most of these models still requires significant computational resources. For the future work, methods that can achieve more lightweight and less computationally heavy model should be considered. Also, more refined occupancy voxels can be explored in future work.

Metrics: For 3D occupancy generation, two qualities of the generated result are highly considered: *(I) 3D occupancy voxel* is evaluated by *IOU* and *mIOU* metric as can be seen in DOME [261]. While *(II) Flow Forecasting* is evaluated by extending the *Video Panoptic Quality* [276] as shown in Drive-OccWorld [259].

3) Multi-modal Generation: Multi-modal generation approaches integrate multiple sensor modalities and data types as input, and output multi-modal data that can include camera images, LiDAR point clouds and depth estimation.

HoloDrive [266] introduces a unified framework for joint 2D-3D scene generation, addressing limitations of single-modality approaches. It employs BEV-to-Camera and Camera-to-BEV transformation modules to bridge heterogeneous generative models. Therefore, it ensures consistency between 2D and 3D representations while leveraging both camera images and LiDAR point clouds for the generation of coherent street scenes.

BEVWorld [267] performed world modeling through a unified BEV latent space that integrates multi-modal sensor inputs. The framework comprises a multi-modal tokenizer and a latent BEV sequence diffusion model that encodes multi-modal data into a unified BEV latent space, aligning visual semantics with geometric information in a self-supervised manner.

GEM [270] proposes a versatile framework for generating realistic environments by integrating multi-modal sensor data, including camera images, and depth estimation. It employs a generative model based on a spatial-temporal transformer capable of predicting dynamic scene evolution regarding visual generation and depth estimation.

Metrics: For multi-modal outputs, 3D point clouds prediction related to HoloDrive [266] and BEVWorld [267] is evaluated by the *chamfer distance* [277] metric; depth estimation result related to GEM [270] is evaluated with the *absolute relative error* of [278].

4) Evaluation & Benchmarks: Current benchmarking frameworks provide standardized methods to assess the quality, controllability, and utility of generated scenarios, ensuring that the WMs meet the requirements for AD applications. Current evaluation frameworks mainly focus on visual realism and on the performance of downstream tasks (perception, planning, etc.).

ACT-Bench [271] introduces a standardized framework to quantify action controllability, measuring how well the generated scenarios adhere to specified driving instructions. This benchmarking framework assesses the fidelity of action execution in WM-generated scenarios. DriveArena [272] is a closed-loop generative simulation platform that enables the evaluation of AD systems in dynamic and realistic environments. By simulating continuous interactions between the ego-vehicle and the environment, it bridges the gap between synthetic training and real-world deployment, supporting the iterative refinement of driving policies.

Despite the promising capabilities, WMs for scenario generation face several limitations. Current implementations struggle with physical realism, when modeling complex multi-agent interactions and the physics of the real-world, including the vehicle dynamics and kinematics laws, tire friction, collision forces, and weather effects. The generated scenarios sometimes contain physically implausible elements, such as objects that appear and suddenly disappear. Hence, the surveyed WMs can generate diverse driving scenarios but cannot accurately satisfy the physics laws. This gap can lead to wrong testing results and infeasible scenarios.

Metrics: Existing benchmarks introduce several metrics to evaluate world models. For instance, ACT-Bench [271] proposes *(I) instruction-execution consistency* to quantify the degree of alignment between the given instructions and the executed actions. ACT-Bench also quantifies the *(II) Trajectory Alignment* (TA) to capture the accuracy andvquality of the executed actions. Specifically, TA integrates the *average displacement error* and the *final displacement error* as proposed by Phong et al. [279]. DriveArena [272] uses two metrics: *(I) Predictive Driver Model Score* (PDM Score) [280] and *(II) Arena Driving Score*, which combines the trajectory PDMS with route completion.

VIII. DATASETS, SIMULATORS AND BENCHMARK CHALLENGES

In this section, we review datasets, simulation platforms, and benchmark challenges that serve as the foundation for scenario generation and analysis with FMs. We intentionally limit our scope to the most recent and impactful resources that are relevant for FM applications, and omit entries already covered in previous work.

A. Datasets

For Language Foundation Models, a common approach is to reproduce real-world scenarios in a simulation environment, and reconstruct the corresponding events. LLMs typically use agents' trajectory data from the provided datasets, while VLMs or MLLMs can leverage additional input modalities such as LiDAR point clouds, RGB images or video streams, and rich annotations. Specifically, DMs use inputs such as RGB images, trajectories, and potentially LiDAR data to generate realistic future scenes or motion patterns through iterative refinement. In contrast, WMs aim to learn the underlying dynamics of driving environments by encoding multimodal sensor data (e.g., images, LiDAR, trajectories) and predicting future states or scene evolutions. Meanwhile, for scenario analysis, a common approach is to leverage VLMs or MLLMs to analyze driving scenes—using image or video data, with or without LiDAR or HD maps—to assess the driving status across different tasks such as perception, prediction, and reasoning.

To assess the relevance and applicability of datasets, we adopt the categorization scheme introduced by Ding et al. [53]. This scheme enables a structured comparison across datasets, considering their sensor coverage, annotation depth, scene diversity, and potential for controllable generative tasks. In the context of FMs, which require large, diverse, and annotated data, the choice of dataset properties is fundamental to enhance the model's generalization potential. We apply this categorization to a selection of impactful and most recent datasets in Table X, using [53] to categorize the dataset's properties given below.

(1) Sensor Data: High-quality datasets like Waymo [72] and nuScenes [121] offer diverse sensor modalities including RGB cameras, LiDAR, and RADAR. Such multimodal input

TABLE X
OVERVIEW OF IMPACTFUL AND RECENT DATASETS FOR FOUNDATION MODEL-BASED SCENARIO GENERATION AND ANALYSIS.

	Dataset	Year	Real	View	Sensor Data				Annotation			Traffic Condition			
					Image	LiDAR	RADAR	Traj.	3D	2D	Lane	Weather	Time	Region	Jam
Impactful	HighD [64]	2018	✓	BEV	RGB			✓		✓			D	H	✓
	nuScenes [121]	2020	✓	FPV	RGB	✓	✓	✓	✓	✓		✓	D/N	U	
	Waymo Open [72]	2020	✓	FPV	RGB	✓		✓	✓	✓	✓	✓	D/N	U/S	
	DRAMA [181]	2022	✓	FPV	RGB				✓		✓			-	U	✓
Most Recent	Comma2k19 [281]	2019	✓	FPV	RGB				✓	✓				D/N	U/S/R/H	✓
	Toronto3D [282]	2020	✓	BEV	RGB	✓			✓	✓		✓		D/N	U	✓
	A2D2 [283]	2020	✓	FPV	RGB	✓	✓		✓	✓		✓	✓	D	U/S/R/H	✓
	WADS [284]	2020	✓	FPV	RGB	✓	✓		✓	✓			✓	D/N	U/S/R	✓
	SeethroughFog [285]	2020	✓	FPV	RGB	✓	✓		✓	✓		✓	✓	D/N	U/S/R/H	✓
	Leddar PixSet [286]	2021	✓	FPV	RGB	✓			✓	✓	✓		✓	D/N	U/S/R	✓
	ZOD [287]	2022	✓	FPV	RGB	✓	✓		✓	✓	✓	✓	✓	D/N	U/S/R/H	✓
	IDD-3D [288]	2022	✓	FPV	RGB	✓				✓	✓			-	R	✓
	CODA [146]	2022	✓	FPV	RGB	✓	✓		✓	✓	✓	✓	✓	D/N	U/S/R	
	SHIFT [289]	2022	✓	FPV	RGB	✓	✓		✓	✓	✓	✓	✓	D/N	U/S/R/H	✓
	DeepAccident [188]	2023		FPV/BEV	RGB/S	✓				✓	✓	✓	✓	D/N	U/S/R/H	✓
	Dual_Radar [290]	2023	✓	FPV	RGB	✓	✓		✓	✓		✓	✓	D/N	U	
	V2V4Real [291]	2023	✓	FPV	RGB	✓			✓	✓		✓		-	U/S/H	
	SCaRL [292]	2024		FPV/BEV	RGB/S	✓	✓		✓	✓	✓	✓	✓	D/N	U/S/R/H	✓
	MARS [293]	2024	✓	FPV	RGB	✓	✓		✓	✓	✓	✓	✓	D/N	U/S/H	
	Scenes101 [294]	2024	✓	FPV	RGB				✓			✓	✓	D/N	U/S/R/H	
	TruckScenes [295]	2025	✓	FPV	RGB	✓	✓		✓	✓		✓	✓	D/N	H/U	

Impactful: We define a dataset's impact by the number of times it was used—not simply cited—by the papers included in our survey. Using this criterion, the four most impactful papers are associated with the following datasets: nuScenes (52 uses), Waymo Open (19), DRAMA (4), and HighD (3).
View indicates: FPV = First-person View, BEV = Bird's-eye View. **Traffic Condition** includes: D/N = Day/Night; U/S/R/H = Urban/Suburban/Rural/Highway; Jam = presence of traffic congestion.

is especially important for pretraining and aligning LLMs, VLMs, DMs, and WMs across visual and spatial reasoning tasks.

(2) Annotation: These datasets also include detailed 2D and 3D object annotations, lane information, and agent trajectories. This level of semantic and geometric detail supports tasks such as perception, prediction, map-conditioned scenario generation, and safety analysis.

(3) Traffic Condition: Traffic condition describes when and where the data was collected, including time of day (day/night), environment type (urban, suburban, rural, highway), and presence of traffic congestion. These factors affect visibility, traffic flow, road layout, and driving behavior, providing diverse scenarios for evaluating autonomous driving performance.

Datasets such as Waymo Open [72] and nuScenes [121] are particularly widespread in the literature. This is largely due to their real-world fidelity, rich multisensor coverage, and comprehensive annotations, which makes them ideal for training and evaluation of FMs. Additionally, it is worth noting that emerging (Visual) Question Answering datasets relevant to scenario analysis with Language Foundation Models are discussed in Section IV-C1 and Section V-C1.

B. Simulators

Simulation platforms are essential in the development and evaluation pipeline of AD systems. They enable safe and reproducible testing, large-scale scenario generation, and structured benchmarking. For Foundation Model-based scenario generation, simulators are particularly valuable for generating training data, enabling self-supervised pretraining, and facilitating the sim-to-real validation. FM-based scenario generation can be performed by LLMs/VLMs/MLLMs through either API functions or domain-specific languages, allowing automatic script generation and scenario execution. Table XI summarizes the impactful and recent simulation platforms which are relevant to scenario generation and analysis. For the classification and evaluation of the existing simulators, we extend the categorization scheme introduced by Ding et al. [53], focusing on features especially relevant to the development and application of FMs.

(1) Backend: The simulation backend defines the physical and rendering engine used to generate sensor data and simulate interactions. Platforms such as Unreal Engine 4 (UE4) or Unity enable high-fidelity rendering and realistic vehicle dynamics, which are valuable for training perception-driven foundation models. Lightweight or symbolic backends, like SUMO or Nocturne, are useful in large-scale planning and decision-making datasets where rendering realism is less critical.

(2) Realistic Perception: Simulators with realistic perception capabilities provide physics-based sensor outputs, including camera, LiDAR, or radar emulation. Such platforms are crucial for training vision-language foundation models, sensor-fusion backbones, or multimodal world models.

(3) Custom Scenario: The ability to define and customize traffic scenarios is a central requirement for both evaluation and data generation workflows. Particularly for foundation models, automated and diverse scenario creation supports the pretraining of models on rare, safety-critical, or systematically varied interactions. Customization typically includes the placement and behavior of traffic participants, route definitions, or modifications of environmental conditions such as weather and lighting. Simulators like CARLA [50] offer rich APIs for manual customization, enabling users to script complex multi-agent interactions and adjust parameters

TABLE XI
OVERVIEW OF IMPACTFUL AND RECENT SIMULATORS FOR FOUNDATION MODEL-BASED SCENARIO GENERATION AND ANALYSIS.

	Simulator	Year	Backend	Open Source	Realistic Perception	Custom Scenario	Map Source		API Supports			DSL Support
							Real World	Human Design	Python	C++	ROS 2	
Impactful	CARLA [50]	2017	UE4	✓	✓	✓		✓	✓	✓	✓	✓
	SUMO [51]	2018	None	✓		✓	✓	✓	✓	✓		✓
	LGSVL [88]	2020	Unity	✓	✓	✓	✓	✓	✓	✓	✓	✓
	MetaDrive [74]	2021	Panda3D	✓	✓	✓	✓	✓	✓			✓
Most Recent	MATLAB AD Toolbox [296]	2018	MATLAB		✓	✓	✓	✓	✓	✓	✓	✓
	Nvidia Drive Sim [297]	2019	Nvidia Omniverse		✓	✓	✓	✓	✓	✓		
	Vista [298]	2020	None	✓	✓	✓	✓		✓			
	Nuplan [48]	2021	None	✓	✓	✓	✓	✓	✓			
	AWSIM [299]	2021	Unity	✓	✓	✓	✓	✓			✓	✓
	InterSim [300]	2022	None	✓	✓	✓	✓		✓			
	Nocturne [301]	2022	None	✓	✓	✓	✓	✓	✓	✓		
	BeamNG.tech [197]	2022	Soft-body physics		✓	✓		✓	✓		✓	✓
	Waymax [302]	2023	JAX	✓	✓	✓		✓	✓			
	TBSim [303]	2023	None	✓	✓	✓	✓	✓	✓			

Impactful: We identify the impact of a simulator following the same criterion of Table X, based on the number of times the simulator was used—not simply cited—by the papers included in our survey. The most impactful simulators are CARLA (8 uses), MetaDrive (4), LGSVL (3), and SUMO (3).

such as vehicle behavior, density, and even scene appearance. More recently, platforms like BeamNG.tech [197] go a step further by supporting automated scenario generation at scale. This enables the procedural creation and batch testing of varied situations, making it well-suited for training and validating foundation models in closed-loop settings.

(4) Map Source: We differentiate between scenarios based on real-world maps (e.g., OpenStreetMap) and those built from human design. Real-world maps ensure geographic realism and coverage, while human-designed maps enable controlled environments.

(5) API-Supports: API support determines how flexibly simulators can be integrated into training pipelines. Python interfaces are especially useful for data generation and model interaction. Robot Operating System (ROS 2) compatibility allows for testing learned policies in robotics stacks, while C++ APIs provide performance for real-time validation and closed-loop deployment.

(6) Domain-Specific Language (DSL) Support: Some simulators provide DSLs that enable structured, human-readable scenario specification through high-level functions or syntax. These interfaces are especially useful for integrating LLMs/VLMs/MLLMs in automated scenario generation pipelines.

Based on these criteria, two simulators stand out in Table XI as particularly impactful in foundation model research: CARLA [50] and SUMO [51]. Their complementary capabilities make them well-suited to different aspects of scenario generation and evaluation. SUMO, a microscopic traffic simulator, is designed for large-scale traffic modeling and interaction-heavy scenario simulation at the population level. It supports integration with real-world maps via OpenStreetMap, allowing for geographically-accurate traffic flow simulations. These features make it a practical backend for LLMs tasked with generating or editing large-scale traffic configurations using natural language prompts or structured templates.

CARLA, in contrast, is a macroscopic simulator with high-fidelity physics, sensor simulation, and photorealistic rendering. It is widely used for ego-agent policy testing in closed-loop environments. Its integration with platforms like Scenic [96] enables programmatic scenario definition through interpretable formal languages, while its Python API offers fine-grained control over agent behavior, environmental settings, and sensor configurations. These characteristics make CARLA particularly suitable for LLMs, VLMs, and MLLMs in vision-language understanding, closed-loop control, and multimodal reasoning.

C. Challenges and Benchmarks

In addition to static datasets and simulation environments, open challenges and benchmarks have become useful tools to evaluate the performance of FMs. While datasets provide the raw material for training and offline testing, challenges enable comparative analysis across models in a controlled and competitive setting.

To our knowledge, this is the first survey to systematically categorize and compare challenges and benchmarks relevant to scenario generation and analysis. Although many of these challenges originate in other application domains, such as medical imaging, robotics, or general-purpose language understanding, their underlying task structures often align with those found in AD. For example, interpreting sensor input, forecasting agent behavior, making multi-step decisions, or generating new representations (e.g., scenes, trajectories, or instructions) are all core operations in scenario understanding.

Table XII presents a selection of challenges and benchmarks published between 2022 and 2025 while our work features a selective overview. The challenges highlight both direct contributions from autonomous driving, such as the Waymo Open Dataset Challenge [306], the Argoverse 2 Scenario Mining Competition [307], and the Accessibility Vision and Autonomy (AVA) Challenge [309], as well as structurally similar benchmarks from other fields. For example, while the Argoverse 2 challenge already touches on scenario analysis, it has not involved scenario generation yet. In

TABLE XII
OVERVIEW OF FOUNDATION MODEL BENCHMARK CHALLENGES FROM 2022–2025, CATEGORIZED BY CORE CAPABILITIES.

Name	Host	Tasks				
		Perception & Interpretation	Prediction & Planning	Reasoning & Decision	Language Understanding	Creative Generation
Autonomous Driving						
CARLA AD Challenge [304]	CARLA					✓
DRL4Real [305]	ICCV					✓
Waymo Open Dataset Challenge [306]	Waymo / CVPR WAD	✓	✓			✓
Argoverse 2: Scenario Mining Competition [307]	ArgoAI			✓	✓	
Roboflow-20VL [308]	Roboflow-VL / CVPR	✓			✓	
AVA Challenge [309]	AVA Challenge Team	✓	✓	✓	✓	
Other Fields Related to Generation and Analysis						
IGLU Challenge [310]	NeurIPS / IGLU Team		✓	✓	✓	
LLM Efficiency Challenge [311]	NeurIPS				✓	
MMWorld [312]	CVPR			✓		
3D Scene Understanding [313]	CVPR	✓			✓	
Trojan Detection [314]	NeurIPS / CAIS				✓	
SMART-101 [315]	CVPR	✓		✓	✓	
NICE Challenge [316]	CVPR / LG Research	✓		✓	✓	
SyntaGen [317]	CVPR	✓				✓
Habitat Challenge [318]	CVPR / FAIR	✓	✓	✓		
BIG-bench [319]	Google Research			✓	✓	
BIG-bench Hard (BBH) [320]	Google Research			✓	✓	
HELM [321]	Stanford CRFM			✓	✓	
MMBench [322]	OpenCompass	✓		✓	✓	
MMMU [323]	CVPR / U-Waterloo / OSU	✓		✓	✓	
Open LLM Leaderboard [324]	VILA-Lab				✓	
Text-to-Image Leaderboard [325]	Artificial Analysis				✓	✓
Ego4D [326]	FAIR	✓	✓	✓	✓	
VizWiz Grand Challenge [327]	CVPR VizWiz Workshop	✓			✓	
MedFM [328]	NeurIPS / Shanghai AI Laboratory	✓				

contrast, tasks such as visual question answering, egocentric video understanding, or synthetic image generation often require models to interpret complex scenes and produce new, coherent outputs, an ability that is equally fundamental for scenario generation. Challenges like SyntaGen [317] and the Text-to-Image Leaderboard [325] illustrate this parallel particularly well: models are asked to generate synthetic examples that exhibit structural realism and diversity. Each challenge is categorized along five core capabilities:

(1) Perception & Interpretation: This category refers to the model's ability to process sensor inputs and extract meaningful semantic representations. Benchmarks such as MMBench [322] and MMMU [323] require fine-grained visual understanding across diagrams, images, and structured visual data. The MedFM [328] challenge focuses on extracting clinically relevant patterns from medical images such as X-rays and histology slides. Ego4D [326] evaluates perception in the context of egocentric video, where models must interpret long, unstructured streams of first-person footage.

(2) Prediction & Planning: Challenges in this category require models to forecast future events or plan a sequence of actions based on partial observations. The Waymo Open Dataset Challenge [306] is a prominent example, assessing motion forecasting from multi-agent sensor streams in real-world traffic scenarios. In the Habitat challenge [318], embodied agents must navigate photo-realistic indoor environments toward semantic or visual goals.

(3) Reasoning & Decision Making: This capability includes commonsense reasoning, causal inference, and multi-hop planning. The BIG-bench [319] and BIG-bench Hard (BBH) [320] benchmarks target difficult problems in logic, mathematics, and abstract reasoning, many

of which remain unsolved even by large-scale models. SMART-101 [315] evaluates reasoning in dialogue, specifically whether models can generate helpful, honest, and harmless responses.

(4) Language Understanding & Generation: This encompasses tasks such as instruction following, question answering, summarization, and dialogue generation. The LLM Efficiency Challenge [311] evaluates how well FMs can be fine-tuned under strict computational budgets. HELM [321] offers a multi-dimensional evaluation across more than a dozen application domains, measuring not only task performance but also fairness, bias, and calibration. The Open LLM Leaderboard [324] provides a public ranking of open-source language models based on standardized evaluations across tasks such as QA, summarization, and reasoning.

(5) Creative Generation: Finally, this category captures the ability of a model to generate complex artifacts such as images, captions, or synthetic data samples. The Text-to-Image Leaderboard [325] evaluates diffusion-based generative models using human preference judgments over image outputs. SyntaGen [317] tests whether diffusion models can generate synthetic images that preserve sufficient structure and diversity to train robust perception models.

Overall, these benchmarks provide a structured landscape for measuring and comparing the capabilities of foundation models beyond narrow task metrics. They reflect the growing demand for models that are not only accurate but also general, flexible, and aligned with real-world deployment requirements, particularly in domains such as autonomous driving and simulation-based scenario generation.

IX. OPEN RESEARCH QUESTIONS AND CHALLENGES

In this paper, we illustrated that the state of the art in the emerging field of scenario generation and analysis with FMs is quite extensive. Nevertheless, there are still some open research questions and challenges. Here, we present a list of open challenges based on additional discussions with leading researchers and experts in the field. These challenges open new research questions to use FMs for scenario generation and analysis in AD.

Challenge 1 – Balancing Plausibility and Edge Case Generation: Effective scenario generation requires balancing realism with the ability to capture rare edge cases. Realistic scenarios demand that FMs abstractly understand the real-world dynamics [329]. On the other hand, edge cases—essential for safety assurance [330]—often approach the boundary of perceived plausibility, making them challenging for FMs to generate without producing unrealistic outcomes. When the plausibility of the generated scenarios is compromised, the resulting scenarios cannot support safety assurance arguments [52]. Thus, the key challenge is ensuring the realism of the generated scenarios, while enabling FMs to generalize and capture critical edge-case situations.

Challenge 2 – Multimodal Data Availability: Many FMs rely on existing datasets that struggle to capture the diversity of real-world driving scenarios. Also, the integration of multimodal data, such as LiDAR, cameras, RADAR, and text, remains limited when compared to single-modality FMs. Indeed, LiDAR and RADAR open-source data are not available for internet-scale training, and domain-specific multimodal datasets still lack scale [41]. Additionally, open-source data covering rare, safety-critical events remains scarce. Hence, a key challenge is the limited availability of rich, multimodal data to support scenario generation with sufficient realism, complexity, and contextual accuracy.

Challenge 3 – Lack of Evaluation & Benchmarks: Currently, scenario evaluation lacks standardization; no widely accepted metrics exist to assess realism, task relevance, or safety-criticality. This limits a meaningful comparison between methods. The open challenges are developing comprehensive, multi-dimensional evaluation metrics and establishing dedicated benchmarks for automated scenario generation and analysis.

Challenge 4 – Safety, Robustness & Verification: Most existing methods lack formal guarantees for safety, correctness, or scenario coverage. The stochastic nature of FMs increases the risk of hallucinated outputs, limiting their reliability for AD safety assurance. A key challenge is ensuring that the generated scenarios are logically grounded—validated through formal verification, constraint satisfaction, or logic-based safety rules—rather than merely correlated with the intended context.

Challenge 5 – Computational Efficiency and Scalability: Current FM-based generation methods demand substantial computational resources, with training requiring massive datasets, long runtimes, and high-performance hardware. Even inference and model fine-tuning are costly without advanced infrastructure. This raises unsolved challenges in scalability, accessibility, and cost-effectiveness, particularly for smaller organizations or resource-constrained applications.

Challenge 6 – Industrial Transferability and Validation: While academia offers many methods for virtual testing and evaluation, the industry must ultimately adapt them for real-world AD applications. Bridging this gap requires method validation, standardization [331], and seamless integration into existing workflows. Thus, a key research question lies in developing approaches that are not only theoretically sound, but also practical, efficient, and accessible to diverse stakeholders—backed by robust industrial validation demonstrating clear benefits and adaptability.

X. FUTURE DIRECTIONS

Addressing the above-mentioned challenges in scenario generation and analysis using FMs yields several directions for future improvement and new research agendas.

Research Direction 1 – Improve Realism: Improving the realism and plausibility of the generated scenarios will require integrating domain-specific knowledge into FMs, enhancing their understanding of real-world dynamics and interactions. Hybrid approaches that combine physics-based models with data-driven FMs offer promise to generate physically-coherent scenarios. Also, the exploration of dreaming with WMs [33] can address gaps in sensor simulation: the data-driven nature of dreaming can capture fine-grained sensor characteristics with high fidelity.

Research Direction 2 – Create Rare Events: Capturing rare, high-risk events requires dedicated methods to systematically identify and generate such scenarios. We recommend creating targeted datasets focused on infrequent but critical situations to improve the accuracy of the models in such cases. Additionally, incorporating reasoning techniques—such as causal or counterfactual reasoning [332]—may help FMs deduce plausible yet uncommon scenarios.

Research Direction 3 – Create Multimodal Datasets: Multimodal data integration remains a major challenge, requiring large-scale datasets specifically designed for scenario generation. These should combine vehicle sensor data—LiDAR, RADAR, cameras—with map data, traffic rules, control actions, human feedback, and textual annotations. We also recommend developing new model architectures and training methods tailored to multimodal fusion to address the current limitations in scalability and integration.

Research Direction 4 – Develop Metrics and KPIs for Comparison: We heavily recommend the development of standardized evaluation methods for an objective comparison of scenarios and scenario generation approaches. This requires new benchmarks and metrics for realism, controllability, diversity, and safety-criticality, along with broad adoption by the community. Promoting these new benchmarks in competitions at the major conferences will drive progress, standardization, and community-driven innovation.

Research Direction 5 – Reduce Computational Demands: Computational efficiency and scalability present major practical constraints. Addressing them requires further investigating techniques such as model distillation, pruning, and quantization, specifically tailored to scenario generation and analysis tasks, to minimize the computational demands without sacrificing performance.

Research Direction 6 – Include Regulatory Compliance: Another research direction is on on how the FM capabilities can be systematically integrated into safety validation workflows within AV development programs. This includes their role in safety data flywheels, where generated scenarios feed into continuous testing, model retraining, safety assessment, and performance monitoring pipelines. We need to focus on ensuring scenario representativeness, balancing real vs synthetic data, aligning with regulatory safety standards, and establishing robust evaluation metrics that capture the safety impact of generated edge cases across the entire AV lifecycle. Additionally, robust methods for data privacy management must be developed to ensure compliance with regulatory standards and ethical norms, safeguarding sensitive information that can be contained in the data learned by the models.

XI. CONCLUSION

This survey examined the state-of-the-art in FMs for autonomous driving applications, emphasizing their significant contributions to both scenario generation and scenario analysis. FMs, including LLMs, VLMs, MLLMs, DMs, and WMs, have emerged as promising tools to enhance the realism, diversity, and scalability of scenario-based testing in AD.

The versatility of FMs lies in their ability to learn from large-scale, heterogeneous datasets through self-supervised training. Their capability to generalize knowledge across various tasks has advanced the scenario-based testing paradigm, overcoming many limitations of traditional rule-based and data-driven methods. Particularly, the dual capability of scenario generation and scenario analysis presented by FMs positions them as crucial enablers for robust and efficient validation frameworks in AD systems.

Despite these advances, notable challenges persist. Achieving fine-grained controllability in safety-critical scenarios, ensuring robust realism in generated scenarios, and addressing computational efficiency remain critical hurdles. Additionally, while the surveyed models demonstrate promising results, further research is needed to enhance the interpretability of their outputs, improve alignment with real-world traffic conditions, and systematically address out-of-distribution scenarios.

Ultimately, as autonomous vehicles approach broader operational domains and higher levels of automation, the role of advanced scenario generation and analysis methods will be paramount. FMs present a powerful framework for this evolution, promising to revolutionize both the safety and efficiency of AD development. The future trajectory of this research is expected to bring further transformative advancements, fostering safer, more reliable, and broadly accessible autonomous mobility.

REFERENCES

[1] J. Betz, M. Lutwitzi, and S. Peters, "A new taxonomy for automated driving: Structuring applications based on their operational design domain, level of automation and automation readiness," in *2024 IEEE Intelligent Vehicles Symposium (IV)*, 2024, pp. 1–7.

[2] Waymo, "Waymo one: The next step on our self-driving journey," 2018. [Online]. Available: https://waymo.com/blog/2018/12/waymo-one-next-step-on-our-self-driving

[3] S. International, "Taxonomy and definitions for terms related to driving automation systems for on-road motor vehicles," *SAE J3016*, 2021. [Online]. Available: https://www.sae.org/standards/content/j3016_202104/

[4] L. Kolodny and J. Elias, "Waymo reports 250,000 paid robotaxi rides per week in u.s." 2025. [Online]. Available: https://www.cnbc.com/2025/04/24/waymo-reports-250000-paid-robotaxi-rides-per-week-in-us.html

[5] S. D. Pendleton, H. Andersen, X. Du, X. Shen, M. Meghjani, Y. H. Eng *et al.*, "Perception, planning, control, and coordination for autonomous vehicles," *Machines*, vol. 5, no. 1, p. 6, 2017.

[6] D. A. Pomerleau, "Alvinn: An autonomous land vehicle in a neural network," *Advances in neural information processing systems*, vol. 1, 1988.

[7] L. Chen, P. Wu, K. Chitta, B. Jaeger, A. Geiger, and H. Li, "End-to-end autonomous driving: Challenges and frontiers," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.

[8] W. Zheng, R. Song, X. Guo, C. Zhang, and L. Chen, "Genad: Generative end-to-end autonomous driving," 2024.

[9] T. Menzel, G. Bagschik, and M. Maurer, "Scenarios for development, test and validation of automated vehicles," in *2018 IEEE intelligent vehicles symposium (IV)*. IEEE, 2018, pp. 1821–1827.

[10] S. Riedmaier, T. Ponn, D. Ludwig, B. Schick, and F. Diermeyer, "Survey on scenario-based safety assessment of automated vehicles," *IEEE access*, vol. 8, pp. 87 456–87 477, 2020.

[11] A. Gambi, V. Nguyen, J. Ahmed, and G. Fraser, "Generating critical driving scenarios from accident sketches," in *2022 IEEE International Conference On Artificial Intelligence Testing (AITest)*. IEEE, 2022, pp. 95–102.

[12] A. Gambi, T. Huynh, and G. Fraser, "Generating effective test cases for self-driving cars from police reports," in *Proceedings of the 2019 27th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, 2019, pp. 257–267.

[13] R. Bommasani, D. A. Hudson, E. Adeli, R. Altman, S. Arora, S. von Arx *et al.*, "On the opportunities and risks of foundation models," *arXiv preprint arXiv:2108.07258*, 2021.

[14] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, 2019, pp. 4171–4186.

[15] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.

[16] J. Jiang, F. Wang, J. Shen, S. Kim, and S. Kim, "A survey on large language models for code generation," *arXiv preprint arXiv:2406.00515*, 2024.

[17] Y. Huang, Y. Chen, and Z. Li, "Applications of large scale foundation models for autonomous driving," *arXiv preprint arXiv:2311.12144*, 2023.

[18] H. Gao, Z. Wang, Y. Li, K. Long, M. Yang, and Y. Shen, "A survey for foundation models in autonomous driving," *arXiv preprint arXiv:2402.01105*, 2024.

[19] Y. Wang, S. Xing, C. Can, R. Li, H. Hua, K. Tian *et al.*, "Generative ai for autonomous driving: Frontiers and opportunities," *arXiv preprint arXiv:2505.08854*, 2025.

[20] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal *et al.*, "Language models are few-shot learners," *Advances in neural information processing systems*, vol. 33, pp. 1877–1901, 2020.

[21] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez *et al.*, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[22] A. Radford, K. Narasimhan, T. Salimans, I. Sutskever *et al.*, "Improving language understanding by generative pre-training," 2018.

[23] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena *et al.*, "Exploring the limits of transfer learning with a unified text-to-text

transformer," *Journal of machine learning research*, vol. 21, no. 140, pp. 1–67, 2020.

[24] A. T. Liu, S.-w. Yang, P.-H. Chi, P.-c. Hsu, and H.-y. Lee, "Mockingjay: Unsupervised speech representation learning with deep bidirectional transformer encoders," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 6419–6423.

[25] P. Yin, G. Neubig, W.-t. Yih, and S. Riedel, "Tabert: Pretraining for joint understanding of textual and tabular data," *arXiv preprint arXiv:2005.08314*, 2020.

[26] H. Luo, L. Ji, B. Shi, H. Huang, N. Duan, T. Li *et al.*, "Univl: A unified video and language pre-training model for multimodal understanding and generation," *arXiv preprint arXiv:2002.06353*, 2020.

[27] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal *et al.*, "Learning transferable visual models from natural language supervision," in *International conference on machine learning*. PmLR, 2021, pp. 8748–8763.

[28] H. Liu, C. Li, Q. Wu, and Y. J. Lee, "Visual instruction tuning," *Advances in neural information processing systems*, vol. 36, pp. 34 892–34 916, 2023.

[29] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *Advances in neural information processing systems*, vol. 33, pp. 6840–6851, 2020.

[30] J. Song, C. Meng, and S. Ermon, "Denoising diffusion implicit models," *arXiv preprint arXiv:2010.02502*, 2020.

[31] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 10 684–10 695.

[32] J. Ho, T. Salimans, A. Gritsenko, W. Chan, M. Norouzi, and D. J. Fleet, "Video diffusion models," *Advances in Neural Information Processing Systems*, vol. 35, pp. 8633–8646, 2022.

[33] D. Ha and J. Schmidhuber, "Recurrent world models facilitate policy evolution," in *Advances in Neural Information Processing Systems 31*. Curran Associates, Inc., 2018, pp. 2451–2463, https://worldmodels.github.io. [Online]. Available: https://papers.nips.cc/paper/7512-recurrent-world-models-facilitate-policy-evolution

[34] X. Wang, Z. Zhu, G. Huang, X. Chen, J. Zhu, and J. Lu, "Drivedreamer: Towards real-world-drive world models for autonomous driving," in *European Conference on Computer Vision*. Springer, 2024, pp. 55–72.

[35] G. Zhao, X. Wang, Z. Zhu, X. Chen, G. Huang, X. Bao *et al.*, "Drivedreamer-2: Llm-enhanced world models for diverse driving video generation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 39, no. 10, 2025, pp. 10 412–10 420.

[36] Y. Wang, S. Xing, C. Can, R. Li, H. Hua, K. Tian *et al.*, "Generative ai for autonomous driving: Frontiers and opportunities," *arXiv preprint arXiv:2505.08854*, 2025.

[37] Y. Zhu, S. Wang, W. Zhong, N. Shen, Y. Li, S. Wang *et al.*, "Will large language models be a panacea to autonomous driving?" *arXiv preprint arXiv:2409.14165*, 2024.

[38] Y. Wu, D. Li, Y. Chen, R. Jiang, H. P. Zou, L. Fang *et al.*, "Multi-agent autonomous driving systems with large language models: A survey of recent advances," *arXiv preprint arXiv:2502.16804*, 2025.

[39] Y. Li, K. Katsumata, E. Javanmardi, and M. Tsukada, "Large language models for human-like autonomous driving: A survey," in *2024 IEEE 27th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2024, pp. 439–446.

[40] X. Zhou, M. Liu, E. Yurtsever, B. L. Zagar, W. Zimmer, H. Cao *et al.*, "Vision language models in autonomous driving: A survey and outlook," *IEEE Transactions on Intelligent Vehicles*, 2024.

[41] C. Cui, Y. Ma, X. Cao, W. Ye, Y. Zhou, K. Liang *et al.*, "A survey on multimodal large language models for autonomous driving," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2024, pp. 958–979.

[42] S. Fourati, W. Jaafar, N. Baccar, and S. Alfattani, "Xlm for autonomous driving systems: A comprehensive review," *arXiv preprint arXiv:2409.10484*, 2024.

[43] J. Li, J. Li, G. Yang, L. Yang, H. Chi, and L. Yang, "Applications of large language models and multimodal large models in autonomous driving: a comprehensive review," *Drones*, 2025.

[44] Y. Guan, H. Liao, Z. Li, J. Hu, R. Yuan, Y. Li *et al.*, "World models for autonomous driving: An initial survey," *IEEE Transactions on Intelligent Vehicles*, 2024.

[45] S. Tu, X. Zhou, D. Liang, X. Jiang, Y. Zhang, X. Li *et al.*, "The role of world models in shaping autonomous driving: A comprehensive survey," *arXiv preprint arXiv:2502.10498*, 2025.

[46] S. Ettinger, S. Cheng, B. Caine, C. Liu, H. Zhao, S. Pradhan *et al.*, "Large scale interactive motion forecasting for autonomous driving: The waymo open motion dataset," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 9710–9719.

[47] B. Wilson, W. Qi, T. Agarwal, J. Lambert, J. Singh, S. Khandelwal *et al.*, "Argoverse 2: Next generation datasets for self-driving perception and forecasting," *arXiv preprint arXiv:2301.00493*, 2023.

[48] N. Karnchanachari, D. Geromichalos, K. S. Tan, N. Li, C. Eriksen, S. Yaghoubi *et al.*, "Towards learning-based planning: The nuplan benchmark for real-world autonomous driving," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 629–636.

[49] M. Althoff, M. Koschi, and S. Manzinger, "Commonroad: Composable benchmarks for motion planning on roads," in *2017 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2017, pp. 719–726.

[50] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "Carla: An open urban driving simulator," in *Conference on robot learning*. PMLR, 2017, pp. 1–16.

[51] P. A. Lopez, E. Wiessner, M. Behrisch, L. Bieker-Walz, J. Erdmann, Y.-P. Flotterod *et al.*, "Microscopic traffic simulation using sumo," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, Nov. 2018, p. 2575–2582.

[52] D. Nalic, T. Mihalj, M. Bäumler, A. Eichberger, and S. Bernsteiner, "Scenario based testing of automated driving systems: A literature survey," in *FISITA web Congress*, vol. 10, 2020, p. 1.

[53] W. Ding, C. Xu, M. Arief, H. Lin, B. Li, and D. Zhao, "A survey on safety-critical driving scenario generation—a methodological perspective," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 7, pp. 6971–6988, 2023.

[54] B. Schütt, J. Ransiek, T. Braun, and E. Sax, "1001 ways of scenario generation for testing of self-driving cars: A survey," in *2023 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2023, pp. 1–8.

[55] Z. Yang, X. Jia, H. Li, and J. Yan, "Llm4drive: A survey of large language models for autonomous driving," *arXiv preprint arXiv:2311.01043*, 2023.

[56] H. Tian, K. Reddy, Y. Feng, M. Quddus, Y. Demiris, and P. Angeloudis, "Large (vision) language models for autonomous vehicles: Current trends and future directions," *Authorea Preprints*, 2024.

[57] A. Fu, Y. Zhou, T. Zhou, Y. Yang, B. Gao, Q. Li *et al.*, "Exploring the interplay between video generation and world models in autonomous driving: A survey," *arXiv preprint arXiv:2411.02914*, 2024.

[58] T. Feng, W. Wang, and Y. Yang, "A survey of world models for autonomous driving," *arXiv preprint arXiv:2501.11260*, 2025.

[59] S. S. Mahmud, L. Ferreira, M. S. Hoque, and A. Tavassoli, "Application of proximal surrogate indicators for safety evaluation: A review of recent developments and research needs," *IATSS research*, vol. 41, no. 4, pp. 153–163, 2017.

[60] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781*, 2013.

[61] J. Kaplan, S. McCandlish, T. Henighan, T. B. Brown, B. Chess, R. Child *et al.*, "Scaling laws for neural language models," *arXiv preprint arXiv:2001.08361*, 2020.

[62] X. Bi, D. Chen, G. Chen, S. Chen, D. Dai, C. Deng *et al.*, "Deepseek llm: Scaling open-source language models with longtermism," *arXiv preprint arXiv:2401.02954*, 2024.

[63] P. Sahoo, A. K. Singh, S. Saha, V. Jain, S. Mondal, and A. Chadha, "A systematic survey of prompt engineering in large language models: Techniques and applications," *arXiv preprint arXiv:2402.07927*, 2024.

[64] R. Krajewski, J. Bock, L. Kloeker, and L. Eckstein, "The highd dataset: A drone dataset of naturalistic vehicle trajectories on german highways for validation of highly automated driving systems," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, Nov. 2018, p. 2118–2125.

[65] C. Chang, D. Cao, L. Chen, K. Su, K. Su, Y. Su *et al.*, "Metascenario: A framework for driving scenario data description, storage and indexing," *IEEE Transactions on Intelligent Vehicles*, vol. 8, no. 2, pp. 1156–1175, 2022.

[66] C. Chang, S. Wang, J. Zhang, J. Ge, and L. Li, "Llmscenario: Large language model driven scenario generation," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2024.

[67] J. Zhang, C. Xu, and B. Li, "Chatscene: Knowledge-enabled safety-critical scenario generation for autonomous vehicles," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 15 459–15 469.

[68] E. Aasi, P. Nguyen, S. Sreeram, G. Rosman, S. Karaman, and D. Rus, "Generating out-of-distribution scenarios using language models," *arXiv preprint arXiv:2411.16554*, 2024.

[69] Y. Mei, T. Nie, J. Sun, and Y. Tian, "Seeking to collide: Online safety-critical scenario generation for autonomous driving with retrieval augmented large language models," *arXiv preprint arXiv:2505.00972*, 2025.

[70] W. Zhan, L. Sun, D. Wang, H. Shi, A. Clausse, M. Naumann *et al.*, "Interaction dataset: An international, adversarial and cooperative motion dataset in interactive driving scenarios with semantic maps," *arXiv preprint arXiv:1910.03088*, 2019.

[71] X. Li, E. Liu, T. Shen, J. Huang, and F.-Y. Wang, "Chatgpt-based scenario engineer: A new framework on scenario generation for trajectory prediction," *IEEE Transactions on Intelligent Vehicles*, vol. 9, no. 3, pp. 4422–4431, 2024.

[72] P. Sun, H. Kretzschmar, X. Dotiwalla, A. Chouard, V. Patnaik, P. Tsui *et al.*, "Scalability in perception for autonomous driving: Waymo open dataset," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 2446–2454.

[73] National Highway Traffic Safety Administration, "National Motor Vehicle Crash Causation Survey (NMVCCS)," https://catalog.data.gov/dataset/national-motor-vehicle-crash-causation-survey-nmvccs, 2024.

[74] Q. Li, Z. Peng, L. Feng, Q. Zhang, Z. Xue, and B. Zhou, "Metadrive: Composing diverse driving scenarios for generalizable reinforcement learning," *IEEE transactions on pattern analysis and machine intelligence*, vol. 45, no. 3, pp. 3461–3475, 2022.

[75] S. Tan, B. Ivanovic, X. Weng, M. Pavone, and P. Kraehenbuehl, "Language conditioned traffic generation," *arXiv preprint arXiv:2307.07947*, 2023.

[76] Y. Zhao, W. Xiao, T. Mihalj, J. Hu, and A. Eichberger, "Chat2scenario: Scenario extraction from dataset through utilization of large language model," in *2024 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2024, pp. 559–566.

[77] OpenStreetMap contributors, "Planet dump retrieved from https://planet.osm.org ," https://www.openstreetmap.org, 2017.

[78] S. Li, T. Azfar, and R. Ke, "Chatsumo: Large language model for automating traffic scenario generation in simulation of urban mobility," *IEEE Transactions on Intelligent Vehicles*, 2024.

[79] B.-K. Ruan, H.-T. Tsui, Y.-H. Li, and H.-H. Shuai, "Traffic scene generation from natural language description for autonomous vehicles with large language model," *arXiv preprint arXiv:2409.09575*, 2024.

[80] A. Aiersilan, "Generating traffic scenarios via in-context learning to learn better motion planner," *arXiv preprint arXiv:2412.18086*, 2024.

[81] S. Tan, B. Ivanovic, Y. Chen, B. Li, X. Weng, Y. Cao *et al.*, "Promptable closed-loop traffic simulation," *arXiv preprint arXiv:2409.05863*, 2024.

[82] Y. Mei, T. Nie, J. Sun, and Y. Tian, "Llm-attacker: Enhancing closed-loop adversarial scenario generation for autonomous driving with large language models," *arXiv preprint arXiv:2501.15850*, 2025.

[83] E. Leurent, "An environment for autonomous driving decision-making," https://github.com/eleurent/highway-env, 2018.

[84] H. Tian, K. Reddy, Y. Feng, M. Quddus, Y. Demiris, and P. Angeloudis, "Enhancing autonomous vehicle training with language model integration and critical scenario generation," *arXiv preprint arXiv:2404.08570*, 2024.

[85] Y. Wei, Z. Wang, Y. Lu, C. Xu, C. Liu, H. Zhao *et al.*, "Editable scene simulation for autonomous driving via collaborative llm-agents," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 15 077–15 087.

[86] Y. Deng, J. Yao, Z. Tu, X. Zheng, M. Zhang, and T. Zhang, "Target: Automated scenario generation from traffic rules for testing autonomous vehicles," *arXiv preprint arXiv:2305.06018*, 2023.

[87] Ç. Güzay, E. Özdemir, and Y. Kara, "A generative ai-driven application: Use of large language models for traffic scenario generation," in *2023 14th International Conference on Electrical and Electronics Engineering (ELECO)*. IEEE, 2023, pp. 1–6.

[88] G. Rong, B. H. Shin, H. Tabatabaee, Q. Lu, S. Lemke, M. Možeiko *et al.*, "Lgsvl simulator: A high fidelity simulator for autonomous driving," in *2020 IEEE 23rd International conference on intelligent transportation systems (ITSC)*. IEEE, 2020, pp. 1–6.

[89] A. Guo, Y. Zhou, H. Tian, C. Fang, Y. Sun, W. Sun *et al.*, "Sovar: Build generalizable scenarios from accident reports for autonomous driving testing," in *Proceedings of the 39th IEEE/ACM International Conference on Automated Software Engineering*, 2024, pp. 268–280.

[90] S. Tang, Z. Zhang, J. Zhou, L. Lei, Y. Zhou, and Y. Xue, "Legend: A top-down approach to scenario generation of autonomous driving systems assisted by large language models," in *Proceedings of the 39th IEEE/ACM International Conference on Automated Software Engineering*, 2024, pp. 1497–1508.

[91] N. Petrovic, K. Lebioda, V. Zolfaghari, A. Schamschurko, S. Kirchner, N. Purschke *et al.*, "Llm-driven testing for autonomous driving scenarios," in *2024 2nd International Conference on Foundation and Large Language Models (FLLM)*. IEEE, 2024, pp. 173–178.

[92] X. Cai, X. Bai, Z. Cui, D. Xie, D. Fu, H. Yu *et al.*, "Text2scenario: Text-driven scenario generation for autonomous driving test," *arXiv preprint arXiv:2503.02911*, 2025.

[93] X. Zhou, Y. Huang, J. Zhang, J. Shao, D. Pan, and P. Li, "Automatic generation method for autonomous driving simulation scenarios based on large language model," in *International Conference on Artificial Intelligence and Autonomous Transportation*. Springer, 2024, pp. 81–91.

[94] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang *et al.*, "Lora: Low-rank adaptation of large language models." *ICLR*, vol. 1, no. 2, p. 3, 2022.

[95] LangChain, "Langchain: The llm application framework," https://github.com/langchain-ai/langchain, 2023, accessed: 2025-05-26.

[96] D. J. Fremont, T. Dreossi, S. Ghosh, X. Yue, A. L. Sangiovanni-Vincentelli, and S. A. Seshia, "Scenic: a language for scenario specification and scene generation," in *Proceedings of the 40th ACM SIGPLAN conference on programming language design and implementation*, 2019, pp. 63–78.

[97] ASAM e.V., "Asam openscenario® dsl v2.1.0," 2024. [Online]. Available: https://www.asam.net/standards/detail/openscenario-dsl/

[98] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix *et al.*, "Llama: Open and efficient foundation language models," 2023.

[99] L. Chen, O. Sinavski, J. Hünermann, A. Karnsund, A. J. Willmott, D. Birch *et al.*, "Driving with llms: Fusing object-level vector modality for explainable autonomous driving," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 14 093–14 100.

[100] C. Schicktanz, L. Klitzke, K. Gimm, G. Rizzo, K. Liesner, H. H. Mosebach *et al.*, "The dlr urban traffic dataset (dlr-ut): A comprehensive traffic dataset from the aim research intersection," *TechRxiv*, 2025.

[101] X. Luo, C. Liu, F. Ding, F. Yang, Y. Zhou, J. Loo *et al.*, "Senserag: Constructing environmental knowledge bases with proactive querying for llm-based autonomous driving," in *Proceedings of the Winter Conference on Applications of Computer Vision*, 2025, pp. 989–996.

[102] A. Elhafsi, R. Sinha, C. Agia, E. Schmerling, I. A. Nesnas, and M. Pavone, "Semantic anomaly detection with large language models," *Autonomous Robots*, vol. 47, no. 8, pp. 1035–1055, 2023.

[103] C. Lu, T. Yue, and S. Ali, "Deepscenario: An open driving scenario dataset for autonomous driving system testing," in *2023 IEEE/ACM 20th International Conference on Mining Software Repositories (MSR)*. IEEE, 2023, pp. 52–56.

[104] J. Wu, C. Lu, A. Arrieta, T. Yue, and S. Ali, "Reality bites: Assessing the realism of driving scenarios with large language models," in *Proceedings of the 2024 IEEE/ACM First International Conference on AI Foundation Models and Software Engineering*, 2024, pp. 40–51.

[105] Y. Gao, M. Piccinini, and J. Betz, "Risk-aware driving scenario analysis with large language models," *arXiv preprint arXiv:2502.02145*, 2025.

[106] S. You, X. Luo, X. Liang, J. Yu, C. Zheng, and J. Gong, "A comprehensive llm-powered framework for driving intelligence evaluation," *arXiv preprint arXiv:2503.05164*, 2025.

[107] C. Jia, Y. Yang, Y. Xia, Y.-T. Chen, Z. Parekh, H. Pham *et al.*, "Scaling up visual and vision-language representation learning with noisy text supervision," in *International conference on machine learning*. PMLR, 2021, pp. 4904–4916.

[108] J. Li, D. Li, C. Xiong, and S. Hoi, "Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation," in *International conference on machine learning*. PMLR, 2022, pp. 12 888–12 900.

[109] J.-B. Alayrac, J. Donahue, P. Luc, A. Miech, I. Barr, Y. Hasson *et al.*, "Flamingo: a visual language model for few-shot learning," *Advances in neural information processing systems*, vol. 35, pp. 23 716–23 736, 2022.

[110] A. Ramesh, M. Pavlov, G. Goh, S. Gray, C. Voss, A. Radford *et al.*, "Zero-shot text-to-image generation," in *International conference on machine learning*. Pmlr, 2021, pp. 8821–8831.

[111] J. Li, D. Li, S. Savarese, and S. Hoi, "Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models," in *International conference on machine learning*. PMLR, 2023, pp. 19 730–19 742.

[112] Y. Inoue, Y. Yada, K. Tanahashi, and Y. Yamaguchi, "Nuscenes-mqa: Integrated evaluation of captions and qa for autonomous driving datasets using markup annotations," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2024, pp. 930–938.

[113] S. Wang, Z. Yu, X. Jiang, S. Lan, M. Shi, N. Chang *et al.*, "Omnidrive: A holistic llm-agent framework for autonomous driving with 3d perception, reasoning and planning," *arXiv preprint arXiv:2405.01533*, 2024.

[114] M. Nie, R. Peng, C. Wang, X. Cai, J. Han, H. Xu *et al.*, "Reason2drive: Towards interpretable and chain-based reasoning for autonomous driving," in *European Conference on Computer Vision*. Springer, 2024, pp. 292–308.

[115] A. Gopalkrishnan, R. Greer, and M. Trivedi, "Multi-frame, lightweight & efficient vision-language models for question answering in autonomous driving," *arXiv preprint arXiv:2403.19838*, 2024.

[116] Z. Sheng, Z. Huang, Y. Qu, Y. Leng, S. Bhavanam, and S. Chen, "Curriculm: Towards safe autonomous driving via personalized safety-critical curriculum learning with vision-language models," *arXiv preprint arXiv:2502.15119*, 2025.

[117] Q. Lu, X. Wang, Y. Jiang, G. Zhao, M. Ma, and S. Feng, "Multimodal large language model driven scenario testing for autonomous vehicles," *arXiv preprint arXiv:2409.06450*, 2024.

[118] A. Marathe, D. Ramanan, R. Walambe, and K. Kotecha, "Wedge: A multi-weather autonomous driving dataset built from generative vision-language models," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2023, pp. 3318–3327.

[119] W. Bao, Q. Yu, and Y. Kong, "Uncertainty-based traffic accident anticipation with spatio-temporal relational learning," in *Proceedings of the 28th ACM International Conference on Multimedia*, 2020, pp. 2682–2690.

[120] Y. Miao, G. Fainekos, B. Hoxha, H. Okamoto, D. Prokhorov, and S. Mitra, "From dashcam videos to driving simulations: Stress testing automated vehicles against rare events," *arXiv preprint arXiv:2411.16027*, 2024.

[121] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu *et al.*, "nuscenes: A multimodal dataset for autonomous driving," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 11 621–11 631.

[122] S. Luo, Y. Zhang, Y. Deng, and X. Zheng, "From accidents to insights: Leveraging multimodal data for scenario-driven ads testing," *arXiv preprint arXiv:2502.02025*, 2025.

[123] T. Choudhary, V. Dewangan, S. Chandhok, S. Priyadarshan, A. Jain, A. K. Singh *et al.*, "Talk2bev: Language-enhanced bird's-eye view maps for autonomous driving," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 16 345–16 352.

[124] J. Mao, M. Niu, C. Jiang, H. Liang, J. Chen, X. Liang *et al.*, "One million scenes for autonomous driving: Once dataset," *arXiv preprint arXiv:2106.11037*, 2021.

[125] A. Ishaq, J. Lahoud, K. More, O. Thawakar, R. Thawkar, D. Dissanayake *et al.*, "Drivelmm-o1: A step-by-step reasoning dataset and large multimodal model for driving scenario understanding," *arXiv preprint arXiv:2503.10621*, 2025.

[126] C. Sima, K. Renz, K. Chitta, L. Chen, H. Zhang, C. Xie *et al.*, "Drivelm: Driving with graph visual question answering," in *European Conference on Computer Vision*. Springer, 2024, pp. 256–274.

[127] A.-M. Marcu, L. Chen, J. Hünermann, A. Karnsund, B. Hanotte, P. Chidananda *et al.*, "Lingoqa: Visual question answering for autonomous driving," in *European Conference on Computer Vision*. Springer, 2024, pp. 252–269.

[128] T. Qian, J. Chen, L. Zhuo, Y. Jiao, and Y.-G. Jiang, "Nuscenes-qa: A multi-modal visual question answering benchmark for autonomous driving scenario," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 5, 2024, pp. 4542–4550.

[129] B. Khalili and A. W. Smyth, "Autodrive-qa-automated generation of multiple-choice questions for autonomous driving datasets using large vision-language models," *arXiv preprint arXiv:2503.15778*, 2025.

[130] M. Najibi, J. Ji, Y. Zhou, C. R. Qi, X. Yan, S. Ettinger *et al.*, "Unsupervised 3d perception with 2d vision-language distillation for autonomous driving," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 8602–8612.

[131] J. Behley, M. Garbade, A. Milioto, J. Quenzel, S. Behnke, C. Stachniss *et al.*, "Semantickitti: A dataset for semantic scene understanding of lidar sequences," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 9297–9307.

[132] Y. Zhou, L. Cai, X. Cheng, Z. Gan, X. Xue, and W. Ding, "Openannotate3d: Open-vocabulary auto-labeling system for multi-modal 3d data," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 9086–9092.

[133] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson *et al.*, "The cityscapes dataset for semantic urban scene understanding," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 3213–3223.

[134] G. J. Brostow, J. Shotton, J. Fauqueur, and R. Cipolla, "Segmentation and recognition using structure from motion point clouds," in *Computer vision–ECCV 2008: 10th European conference on computer vision, marseille, France, October 12-18, 2008, proceedings, part i 10*. Springer, 2008, pp. 44–57.

[135] W.-B. Kou, Q. Lin, M. Tang, S. Wang, R. Ye, G. Zhu *et al.*, "Enhancing large vision model in street scene semantic understanding through leveraging posterior optimization trajectory," *arXiv preprint arXiv:2501.01710*, 2025.

[136] I. de Zarzà, J. de Curtò, G. Roig, and C. T. Calafate, "Llm multimodal traffic accident forecasting," *Sensors*, vol. 23, no. 22, p. 9225, 2023.

[137] X. Zheng, L. Wu, Z. Yan, Y. Tang, H. Zhao, C. Zhong *et al.*, "Large language models powered context-aware motion prediction in autonomous driving," in *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2024, pp. 980–985.

[138] F. Yu, H. Chen, X. Wang, W. Xian, Y. Chen, F. Liu *et al.*, "Bdd100k: A diverse driving dataset for heterogeneous multitask learning," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 2636–2645.

[139] E. Rivera, J. Lübberstedt, N. Uhlemann, and M. Lienkamp, "Scenario understanding of traffic scenes through large visual language models," *arXiv preprint arXiv:2501.17131*, 2025.

[140] J. Kim, A. Rohrbach, T. Darrell, J. Canny, and Z. Akata, "Textual explanations for self-driving vehicles," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 563–578.

[141] L. Wen, X. Yang, D. Fu, X. Wang, P. Cai, X. Li *et al.*, "On the road with gpt-4v (ision): Early explorations of visual-language model on autonomous driving," *arXiv preprint arXiv:2311.05332*, 2023.

[142] X. Cao, T. Zhou, Y. Ma, W. Ye, C. Cui, K. Tang *et al.*, "Maplm: A real-world large-scale vision-language benchmark for map and traffic scene understanding," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 21 819–21 830.

[143] A. Keskar, S. Peristela, and R. Greer, "Evaluating multimodal vision-language model prompting strategies for visual question answering in road scene understanding," in *Proceedings of the Winter Conference on Applications of Computer Vision (WACV) Workshops*, February 2025, pp. 1027–1036.

[144] S. Xie, L. Kong, Y. Dong, C. Sima, W. Zhang, Q. A. Chen *et al.*, "Are vlms ready for autonomous driving? an empirical study from the reliability, data, and metric perspectives," *arXiv preprint arXiv:2501.04003*, 2025.

[145] F. Li, H. Jin, B. Gao, L. Fan, L. Jiang, and L. Zeng, "Nugrounding: A multi-view 3d visual grounding framework in autonomous driving," *arXiv preprint arXiv:2503.22436*, 2025.

[146] K. Li, K. Chen, H. Wang, L. Hong, C. Ye, J. Han *et al.*, "Coda: A real-world road corner case dataset for object detection in autonomous driving," in *European Conference on Computer Vision*. Springer, 2022, pp. 406–423.

[147] K. Chen, Y. Li, W. Zhang, Y. Liu, P. Li, R. Gao *et al.*, "Automated evaluation of large vision-language models on self-driving corner cases," in *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 2025, pp. 7817–7826.

[148] Y. Wang, A. Alhuraish, S. Yuan, and H. Zhou, "Openlka: An open dataset of lane keeping assist from recent car models under real-world driving conditions," *arXiv preprint arXiv:2505.09092*, 2025.

[149] H. Hwang, S. Kwon, Y. Kim, and D. Kim, "Is it safe to cross? interpretable risk assessment with gpt-4v for safety-aware street crossing," in *2024 21st International Conference on Ubiquitous Robots (UR)*. IEEE, 2024, pp. 281–288.

[150] F.-H. Chan, Y.-T. Chen, Y. Xiang, and M. Sun, "Anticipating accidents in dashcam videos," in *Computer Vision–ACCV 2016: 13th Asian Conference on Computer Vision, Taipei, Taiwan, November 20-24, 2016, Revised Selected Papers, Part IV 13*. Springer, 2017, pp. 136–153.

[151] J. Zhang, Y. Guan, C. Wang, H. Liao, G. Zhang, and Z. Li, "Latte: Lightweight attention-based traffic accident anticipation engine," *arXiv preprint arXiv:2504.04103*, 2025.

[152] M. P. Ronecker, M. Foutter, A. Elhafsi, D. Gammelli, I. Barakaiev, M. Pavone *et al.*, "Vision foundation model embedding-based semantic anomaly detection," *arXiv preprint arXiv:2505.07998*, 2025.

[153] Q. Zhang, M. Zhu, and H. F. Yang, "Think-driver: From driving-scene understanding to decision-making with vision language models," in *European Conference on Computer Vision Workshop*, 2024.

[154] J. Lee, J. Cho, H. Suk, and S. Kim, "SFF rendering-based uncertainty prediction using visionLLM," in *AAAI 2025 Workshop LM4Plan*, 2025. [Online]. Available: https://openreview.net/forum?id=q8ptjh1pDl

[155] D. Chen, Z. Zhang, Y. Liu, and X. T. Yang, "Insight: Enhancing autonomous driving safety through vision-language models on context-aware hazard detection and edge case evaluation," 2025. [Online]. Available: https://arxiv.org/abs/2502.00262

[156] Y. Wang and H. Zhou, "Bridging human oversight and black-box driver assistance: Vision-language models for predictive alerting in lane keeping assist systems," *arXiv preprint arXiv:2505.11535*, 2025.

[157] D. Wu, W. Han, Y. Liu, T. Wang, C.-z. Xu, X. Zhang *et al.*, "Language prompt for autonomous driving," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 39, no. 8, 2025, pp. 8359–8367.

[158] D. Wu, W. Han, T. Wang, X. Dong, X. Zhang, and J. Shen, "Referring multi-object tracking," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2023, pp. 14 633–14 642.

[159] W. Hong, W. Wang, Q. Lv, J. Xu, W. Yu, J. Ji *et al.*, "Cogagent: A visual language model for gui agents," 2024.

[160] H. Zhang, X. Li, and L. Bing, "Video-llama: An instruction-tuned audio-visual language model for video understanding," *arXiv preprint arXiv:2306.02858*, 2023.

[161] D. Zhu, J. Chen, X. Shen, X. Li, and M. Elhoseiny, "Minigpt-4: Enhancing vision-language understanding with advanced large language models," *arXiv preprint arXiv:2304.10592*, 2023.

[162] J. Achiam, S. Adler, S. Agarwal, I. Ahmad, I. Akkaya, F. L. Aleman *et al.*, "Gpt-4 technical report," *arXiv preprint arXiv:2303.08774*, 2023.

[163] G. Team, R. Anil, S. Borgeaud, J.-B. Alayrac, J. Yu, R. Soricut *et al.*, "Gemini: a family of highly capable multimodal models," *arXiv preprint arXiv:2312.11805*, 2023.

[164] J. Bai, S. Bai, Y. Chu, Z. Cui, K. Dang, X. Deng *et al.*, "Qwen technical report," *arXiv preprint arXiv:2309.16609*, 2023.

[165] S.-Y. Park, C. Cui, Y. Ma, A. Moradipari, R. Gupta, K. Han *et al.*, "Nuplanqa: A large-scale dataset and benchmark for multi-view driving scene understanding in multi-modal large language models," *arXiv preprint arXiv:2503.12772*, 2025.

[166] X. Ding, J. Han, H. Xu, X. Liang, W. Zhang, and X. Li, "Holistic autonomous driving understanding by bird's-eye-view injected multi-modal large models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 13 668–13 677.

[167] A. Ishaq, J. Lahoud, F. S. Khan, S. Khan, H. Cholakkal, and R. M. Anwer, "Tracking meets large multimodal models for driving scenario understanding," *arXiv preprint arXiv:2503.14498*, 2025.

[168] S. Yang, J. Liu, R. Zhang, M. Pan, Z. Guo, X. Li *et al.*, "Lidar-llm: Exploring the potential of large language models for 3d lidar understanding," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 39, no. 9, 2025, pp. 9247–9255.

[169] Z. Xu, Y. Zhang, E. Xie, Z. Zhao, Y. Guo, K.-Y. K. Wong *et al.*, "Drivegpt4: Interpretable end-to-end autonomous driving via large language model," *IEEE Robotics and Automation Letters*, 2024.

[170] S. Park, M. Lee, J. Kang, H. Choi, Y. Park, J. Cho *et al.*, "Vlaad: Vision and language assistant for autonomous driving," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2024, pp. 980–987.

[171] X. Ding, J. Han, H. Xu, W. Zhang, and X. Li, "Hilm-d: Towards high-resolution understanding in multimodal large language models for autonomous driving," *arXiv preprint arXiv:2309.05186*, 2023.

[172] Y. Ma, Y. Cao, J. Sun, M. Pavone, and C. Xiao, "Dolphins: Multimodal language model for driving," in *European Conference on Computer Vision*. Springer, 2024, pp. 403–420.

[173] J. Fan, J. Wu, J. Gao, J. Yu, Y. Wang, H. Chu *et al.*, "Mllm-sul: Multimodal large language model for semantic scene understanding and localization in traffic scenarios," *arXiv preprint arXiv:2412.19406*, 2024.

[174] Y. Zhang and Y. Nie, "Interndrive: A multimodal large language model for autonomous driving scenario understanding," in *Proceedings of the 2024 4th International Conference on Artificial Intelligence, Automation and High Performance Computing*, 2024, pp. 294–305.

[175] P. Gao, J. Han, R. Zhang, Z. Lin, S. Geng, A. Zhou *et al.*, "Llama-adapter v2: Parameter-efficient visual instruction model," 2023.

[176] X. Zhou, K. Larintzakis, H. Guo, W. Zimmer, M. Liu, H. Cao *et al.*, "Tumtraffic-videoqa: A benchmark for unified spatio-temporal video understanding in traffic scenes," *arXiv preprint arXiv:2502.02449*, 2025.

[177] Q. Lu, M. Ma, X. Dai, X. Wang, and S. Feng, "Realistic corner case generation for autonomous vehicles with multimodal large language model," *arXiv preprint arXiv:2412.00243*, 2024.

[178] V. Ramanishka, Y.-T. Chen, T. Misu, and K. Saenko, "Toward driving scene understanding: A dataset for learning driver behavior and causal reasoning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 7699–7707.

[179] H. Tian, X. Han, G. Wu, Y. Zhou, S. Li, J. Wei *et al.*, "An llm-enhanced multi-objective evolutionary search for autonomous driving test scenario generation," *arXiv preprint arXiv:2406.10857*, 2024.

[180] Baidu Apollo team, "Apollo: Open Source Autonomous Driving," https://github.com/ApolloAuto/apollo, 2017, accessed: 2019-02-11.

[181] S. Malla, C. Choi, I. Dwivedi, J. H. Choi, and J. Li, "Drama: Joint risk localization and captioning in driving," in *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, 2023, pp. 1043–1052.

[182] J. M. Hankey, M. A. Perez, and J. A. McClafferty, "Description of the shrp 2 naturalistic database and the crash, near-crash, and baseline data sets," Virginia Tech Transportation Institute, Tech. Rep., 2016.

[183] T. Zeng, L. Wu, L. Shi, D. Zhou, and F. Guo, "Are vision llms road-ready? a comprehensive benchmark for safety-critical driving video understanding," *arXiv preprint arXiv:2504.14526*, 2025.

[184] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *2012 IEEE conference on computer vision and pattern recognition*. IEEE, 2012, pp. 3354–3361.

[185] S. Jain, S. Thapa, K.-T. Chen, A. L. Abbott, and A. Sarkar, "Semantic understanding of traffic scenes with large vision language models," in *2024 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2024, pp. 1580–1587.

[186] Q. Kong, Y. Kawana, R. Saini, A. Kumar, J. Pan, T. Gu *et al.*, "Wts: A pedestrian-centric traffic video dataset for fine-grained spatial-temporal understanding," in *European Conference on Computer Vision*. Springer, 2024, pp. 1–18.

[187] J. Lübberstedt, E. Rivera, N. Uhlemann, and M. Lienkamp, "V3lma: Visual 3d-enhanced language model for autonomous driving," *arXiv preprint arXiv:2505.00156*, 2025.

[188] T. Wang, S. Kim, J. Wenxuan, E. Xie, C. Ge, J. Chen *et al.*, "Deepaccident: A motion and accident prediction benchmark for v2x autonomous driving," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 6, 2024, pp. 5599–5606.

[189] L. Wang, Y. Ren, H. Jiang, P. Cai, D. Fu, T. Wang *et al.*, "Accidentgpt: Accident analysis and prevention from v2x environmental perception with multi-modal large model," *arXiv preprint arXiv:2312.13156*, 2023.

[190] L. Shi, B. Jiang, and F. Guo, "Scvlm: a vision-language model for driving safety critical event understanding," *arXiv preprint arXiv:2410.00982*, 2024.

[191] M. Abu Tami, H. I. Ashqar, M. Elhenawy, S. Glaser, and A. Rakotoniriainy, "Using multimodal large language models (mllms) for automated detection of traffic safety-critical events," *Vehicles*, vol. 6, no. 3, pp. 1571–1590, 2024.

[192] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli, "Deep unsupervised learning using nonequilibrium thermodynamics," in *International conference on machine learning*. pmlr, 2015, pp. 2256–2265.

[193] P. Dhariwal and A. Nichol, "Diffusion models beat gans on image synthesis," *Advances in neural information processing systems*, vol. 34, pp. 8780–8794, 2021.

[194] A. Nichol, P. Dhariwal, A. Ramesh, P. Shyam, P. Mishkin, B. McGrew *et al.*, "Glide: Towards photorealistic image generation and editing with text-guided diffusion models," *arXiv preprint arXiv:2112.10741*, 2021.

[195] L. Zhang, A. Rao, and M. Agrawala, "Adding conditional control to text-to-image diffusion models," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2023, pp. 3836–3847.

[196] W. Peebles and S. Xie, "Scalable diffusion models with transformers," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2023, pp. 4195–4205.

[197] P. Maul, M. Mueller, F. Enkler, E. Pigova, T. Fischer, and L. Stamatogiannakis, "Beamng.tech technical paper," BeamNG GmbH, Technical Report, 2023. [Online]. Available: https://beamng.tech/

[198] Z. Zhong, D. Rempe, D. Xu, Y. Chen, S. Veer, T. Che *et al.*, "Guided conditional diffusion for controllable traffic simulation," in *2023 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2023, pp. 3560–3566.

[199] S. Zhang, J. Tian, Z. Zhu, S. Huang, J. Yang, and W. Zhang, "Drivegen: Towards infinite diverse traffic scenarios with large models," *arXiv preprint arXiv:2503.05808*, 2025.

[200] E. Pronovost, M. R. Ganesina, N. Hendy, Z. Wang, A. Morales, K. Wang *et al.*, "Scenario diffusion: Controllable driving scenario generation with diffusion," *Advances in Neural Information Processing Systems*, vol. 36, pp. 68 873–68 894, 2023.

[201] M. Jiang, Y. Bai, A. Cornman, C. Davis, X. Huang, H. Jeon *et al.*, "Scenediffuser: Efficient and controllable driving simulation initialization and rollout," *Advances in Neural Information Processing Systems*, vol. 37, pp. 55 729–55 760, 2024.

[202] S. Sun, Z. Gu, T. Sun, J. Sun, C. Yuan, Y. Han *et al.*, "Drivescenegen: Generating diverse and realistic driving scenarios from scratch," *IEEE Robotics and Automation Letters*, 2024.

[203] K. Chitta, D. Dauner, and A. Geiger, "Sledge: Synthesizing driving environments with generative models and rule-based traffic," in *European Conference on Computer Vision*. Springer, 2024, pp. 57–74.

[204] L. Rowe, R. Girgis, A. Gosselin, L. Paull, C. Pal, and F. Heide, "Scenario dreamer: Vectorized latent diffusion for generating driving simulation environments," *arXiv preprint arXiv:2503.22496*, 2025.

[205] S. Yu, K. Kim, D. Kim, H. Han, and J. Lee, "Direct preference optimization-enhanced multi-guided diffusion model for traffic scenario generation," *arXiv preprint arXiv:2502.12178*, 2025.

[206] H. Lin, X. Huang, T. Phan-Minh, D. S. Hayden, H. Zhang, D. Zhao *et al.*, "Causal composition diffusion model for closed-loop traffic generation," *arXiv preprint arXiv:2412.17920*, 2024.

[207] C. Xu, D. Zhao, A. Sangiovanni-Vincentelli, and B. Li, "Diffscene: Diffusion-based safety-critical scenario generation for autonomous vehicles," in *The Second Workshop on New Frontiers in Adversarial Machine Learning*, 2023.

[208] J. Lu, S. Azam, G. Alcan, and V. Kyrki, "Data-driven diffusion models for enhancing safety in autonomous vehicle traffic simulations," *arXiv preprint arXiv:2410.04809*, 2024.

[209] Y. Xie, X. Guo, C. Wang, K. Liu, and L. Chen, "Advdiffuser: Generating adversarial safety-critical driving scenarios via guided diffusion," in *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2024, pp. 9983–9989.

[210] W.-J. Chang, F. Pittaluga, M. Tomizuka, W. Zhan, and M. Chandraker, "Safe-sim: Safety-critical closed-loop traffic simulation with diffusion-controllable adversaries," in *European Conference on Computer Vision*. Springer, 2024, pp. 242–258.

[211] Z. Huang, Z. Zhang, A. Vaidya, Y. Chen, C. Lv, and J. F. Fisac, "Versatile behavior diffusion for generalized traffic agent simulation," *arXiv preprint arXiv:2404.02524*, 2024.

[212] Z. Zhong, D. Rempe, Y. Chen, B. Ivanovic, Y. Cao, D. Xu *et al.*, "Language-guided traffic simulation via scene-level diffusion," in *Conference on Robot Learning*. PMLR, 2023, pp. 144–177.

[213] M. Peng, Y. Xie, X. Guo, R. Yao, H. Yang, and J. Ma, "Ld-scene: Llm-guided diffusion for controllable generation of adversarial safety-critical driving scenarios," *arXiv preprint arXiv:2505.11247*, 2025.

[214] J. Zhou, L. Wang, Q. Meng, and X. Wang, "Diffroad: Realistic and diverse road scenario generation for autonomous vehicle testing," *arXiv preprint arXiv:2411.09451*, 2024.

[215] E. Pronovost, K. Wang, and N. Roy, "Generating driving scenes with diffusion," *arXiv preprint arXiv:2305.18452*, 2023.

[216] J. Lu, K. Wong, C. Zhang, S. Suo, and R. Urtasun, "Scenecontrol: Diffusion for controllable traffic scene generation," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 16 908–16 914.

[217] S. Gu, J. Su, Y. Duan, X. Chen, J. Luo, and H. Zhao, "Text2street: Controllable text-to-image generation for street views," in *International Conference on Pattern Recognition*. Springer, 2025, pp. 130–145.

[218] K. Chen, E. Xie, Z. Chen, Y. Wang, L. Hong, Z. Li *et al.*, "Geodiffusion: Text-prompted geometric control for object detection data generation," *arXiv preprint arXiv:2306.04607*, 2023.

[219] K. Yang, E. Ma, J. Peng, Q. Guo, D. Lin, and K. Yu, "Bevcontrol: Accurately controlling street-view elements with multi-perspective consistency via bev sketch layout," *arXiv preprint arXiv:2308.01661*, 2023.

[220] R. Gao, K. Chen, E. Xie, L. Hong, Z. Li, D.-Y. Yeung *et al.*, "Magicdrive: Street view generation with diverse 3d geometry control," *arXiv preprint arXiv:2310.02601*, 2023.

[221] H. Li, Z. Yang, Z. Qian, G. Zhao, Y. Huang, J. Yu *et al.*, "Dualdiff: Dual-branch diffusion model for autonomous driving with semantic fusion," *arXiv preprint arXiv:2505.01857*, 2025.

[222] Y. Wen, Y. Zhao, Y. Liu, F. Jia, Y. Wang, C. Luo *et al.*, "Panacea: Panoramic and controllable video generation for autonomous driving," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 6902–6912.

[223] X. Li, Y. Zhang, and X. Y. Drivingdiffusion, "Layout-guided multi-view driving scene video generation with latent diffusion model," *arXiv preprint arXiv:2310.07771*, vol. 2, no. 3, 2023.

[224] J. Jiang, G. Hong, M. Zhang, H. Hu, K. Zhan, R. Shao *et al.*, "Dive: Efficient multi-view driving scenes generation based on video diffusion transformer," *arXiv preprint arXiv:2504.19614*, 2025.

[225] X. Bai, Y. Luo, L. Jiang, and S. Ostadabbas, "Dual-conditioned temporal diffusion modeling for long driving video generation," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2025.

[226] Y. Fu, A. Jain, X. Chen, Z. Mo, and X. Di, "Drivegenvlm: Real-world video generation for vision language model based autonomous driving," in *2024 IEEE International Automated Vehicle Validation Conference (IAVVC)*. IEEE, 2024, pp. 1–6.

[227] Z. Yang, Z. Qian, X. Li, W. Xu, G. Zhao, R. Yu *et al.*, "Dualdiff+: Dual-branch diffusion for high-fidelity video generation with reward guidance," *arXiv preprint arXiv:2503.03689*, 2025.

[228] Y. Fu, Y. Li, and X. Di, "Gendds: Generating diverse driving video scenarios with prompt-to-video generative model," *arXiv preprint arXiv:2408.15868*, 2024.

[229] Y. Yao, X. Wang, M. Xu, Z. Pu, Y. Wang, E. Atkins *et al.*, "Dota: Unsupervised detection of traffic anomaly in driving videos," *IEEE transactions on pattern analysis and machine intelligence*, vol. 45, no. 1, pp. 444–459, 2022.

[230] Z. Guo, Y. Zhou, and C. Gou, "Drivinggen: Efficient safety-critical driving video generation with latent diffusion models," in *2024 IEEE International Conference on Multimedia and Expo (ICME)*. IEEE, 2024, pp. 1–6.

[231] J. Fang, L.-l. Li, J. Zhou, J. Xiao, H. Yu, C. Lv *et al.*, "Abductive ego-view accident video understanding for safe driving perception," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 22 030–22 040.

[232] C. Li, K. Zhou, T. Liu, Y. Wang, M. Zhuang, H.-a. Gao *et al.*, "Avd2: Accident video diffusion for accident video description," *arXiv preprint arXiv:2502.14801*, 2025.

[233] L. Baresi, D. Y. X. Hu, A. Stocco, and P. Tonella, "Efficient domain augmentation for autonomous driving testing using diffusion models," in *Proceedings of 47th International Conference on Software Engineering*, ser. ICSE '25. IEEE, 2025.

[234] D. Podell, Z. English, K. Lacey, A. Blattmann, T. Dockhorn, J. Müller *et al.*, "SDXL: Improving latent diffusion models for high-resolution image synthesis," in *The Twelfth International Conference on Learning Representations*, 2024. [Online]. Available: https://openreview.net/forum?id=di52zR8xgf

[235] J. Mullan, D. Crawbuck, and A. Sastry, "Hotshot-xl," https://github.com/hotshotco/hotshot-xl, October 2023, online.

[236] T. Brooks, B. Peebles, C. Holmes, W. DePue, Y. Guo, L. Jing *et al.*, "Video generation models as world simulators," *OpenAI Blog*, vol. 1, p. 8, 2024.

[237] Y. LeCun, "A path towards autonomous machine intelligence version 0.9. 2, 2022-06-27," *Open Review*, vol. 62, no. 1, pp. 1–62, 2022.

[238] Y. Wang, J. He, L. Fan, H. Li, Y. Chen, and Z. Zhang, "Driving into the future: Multiview visual forecasting and planning with world model for autonomous driving," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 14 749–14 759.

[239] A. Hu, L. Russell, H. Yeo, Z. Murez, G. Fedoseev, A. Kendall *et al.*, "Gaia-1: A generative world model for autonomous driving," *arXiv preprint arXiv:2309.17080*, 2023.

[240] F. Jia, W. Mao, Y. Liu, Y. Zhao, Y. Wen, C. Zhang *et al.*, "Adriver-i: A general world model for autonomous driving," *arXiv preprint arXiv:2311.13549*, 2023.

[241] L. Russell, A. Hu, L. Bertoni, G. Fedoseev, J. Shotton, E. Arani *et al.*, "Gaia-2: A controllable multi-view generative world model for autonomous driving," *arXiv preprint arXiv:2503.20523*, 2025.

[242] A. Blattmann, T. Dockhorn, S. Kulal, D. Mendelevitch, M. Kilian, D. Lorenz *et al.*, "Stable video diffusion: Scaling latent video diffusion models to large datasets," *arXiv preprint arXiv:2311.15127*, 2023.

[243] G. Zhao, C. Ni, X. Wang, Z. Zhu, X. Zhang, Y. Wang *et al.*, "Drivedreamer4d: World models are effective data machines for 4d driving scene representation," *arXiv preprint arXiv:2410.13571*, 2024.

[244] J. Yang, S. Gao, Y. Qiu, L. Chen, T. Li, B. Dai *et al.*, "Generalized predictive model for autonomous driving," in *Proceedings of the*

IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2024, pp. 14 662–14 672.

[245] S. Gao, J. Yang, L. Chen, K. Chitta, Y. Qiu, A. Geiger *et al.*, "Vista: A generalizable driving world model with high fidelity and versatile controllability," *arXiv preprint arXiv:2405.17398*, 2024.

[246] X. Hu, W. Yin, M. Jia, J. Deng, X. Guo, Q. Zhang *et al.*, "Drivingworld: Constructingworld model for autonomous driving via video gpt," *arXiv preprint arXiv:2412.19505*, 2024.

[247] A. Blattmann, R. Rombach, H. Ling, T. Dockhorn, S. W. Kim, S. Fidler *et al.*, "Align your latents: High-resolution video synthesis with latent diffusion models," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2023, pp. 22 563–22 575.

[248] Y. Wang, J. He, L. Fan, H. Li, Y. Chen, and Z. Zhang, "Driving into the future: Multiview visual forecasting and planning with world model for autonomous driving," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 14 749–14 759.

[249] R. Gao, K. Chen, Z. Li, L. Hong, Z. Li, and Q. Xu, "Magicdrive3d: Controllable 3d generation for any-view rendering in street scenes," *arXiv preprint arXiv:2405.14475*, 2024.

[250] R. Gao, K. Chen, B. Xiao, L. Hong, Z. Li, and Q. Xu, "Magicdriveedit: High-resolution long video generation for autonomous driving with adaptive control," *arXiv preprint arXiv:2411.13807*, 2024.

[251] X. Tian, T. Jiang, L. Yun, Y. Mao, H. Yang, Y. Wang *et al.*, "Occ3d: A large-scale 3d occupancy prediction benchmark for autonomous driving," *Advances in Neural Information Processing Systems*, vol. 36, pp. 64 318–64 330, 2023.

[252] J. Lu, Z. Huang, Z. Yang, J. Zhang, and L. Zhang, "Wovogen: World volume-aware diffusion for controllable multi-camera driving scene generation," in *European Conference on Computer Vision*. Springer, 2024, pp. 329–345.

[253] C. Ni, G. Zhao, X. Wang, Z. Zhu, W. Qin, G. Huang *et al.*, "Recondreamer: Crafting world models for driving scene reconstruction via online restoration," *arXiv preprint arXiv:2411.19548*, 2024.

[254] N. Agarwal, A. Ali, M. Bala, Y. Balaji, E. Barker, T. Cai *et al.*, "Cosmos world foundation model platform for physical ai," *arXiv preprint arXiv:2501.03575*, 2025.

[255] H. A. Alhaija, J. Alvarez, M. Bala, T. Cai, T. Cao, L. Cha *et al.*, "Cosmos-transfer1: Conditional world generation with adaptive multimodal control," *arXiv preprint arXiv:2503.14492*, 2025.

[256] A. Chen, W. Zheng, Y. Wang, X. Zhang, K. Zhan, P. Jia *et al.*, "Geodrive: 3d geometry-informed driving world model with precise action control," *arXiv preprint arXiv:2505.22421*, 2025.

[257] L. Wang, W. Zheng, Y. Ren, H. Jiang, Z. Cui, H. Yu *et al.*, "Occsora: 4d occupancy generation models as world simulators for autonomous driving," *arXiv preprint arXiv:2405.20337*, 2024.

[258] J. Houston, G. Zuidhof, L. Bergamini, Y. Ye, L. Chen, A. Jain *et al.*, "One thousand and one hours: Self-driving motion prediction dataset," in *Conference on Robot Learning*. PMLR, 2021, pp. 409–418.

[259] Y. Yang, J. Mei, Y. Ma, S. Du, W. Chen, Y. Qian *et al.*, "Driving in the occupancy world: Vision-centric 4d occupancy forecasting and planning via world models for autonomous driving," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 39, no. 9, 2025, pp. 9327–9335.

[260] X. Ma, Y. Wang, G. Jia, X. Chen, Z. Liu, Y.-F. Li *et al.*, "Latte: Latent diffusion transformer for video generation," *arXiv preprint arXiv:2401.03048*, 2024.

[261] S. Gu, W. Yin, B. Jin, X. Guo, J. Wang, H. Li *et al.*, "Dome: Taming diffusion model into high-fidelity controllable occupancy world model," *arXiv preprint arXiv:2410.10429*, 2024.

[262] Z. Yan, W. Dong, Y. Shao, Y. Lu, L. Haiyang, J. Liu *et al.*, "Renderworld: World model with self-supervised 3d label," *arXiv preprint arXiv:2409.11356*, 2024.

[263] J. Wei, S. Yuan, P. Li, Q. Hu, Z. Gan, and W. Ding, "Occllama: An occupancy-language-action generative world model for autonomous driving," *arXiv preprint arXiv:2409.03272*, 2024.

[264] O. Contributors, "Openscene: The largest up-to-date 3d occupancy prediction benchmark in autonomous driving," https://github.com/OpenDriveLab/OpenScene, 2023.

[265] C. Min, D. Zhao, L. Xiao, J. Zhao, X. Xu, Z. Zhu *et al.*, "Driveworld: 4d pre-trained scene understanding via world models for autonomous driving," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 15 522–15 533.

[266] Z. Wu, J. Ni, X. Wang, Y. Guo, R. Chen, L. Lu *et al.*, "Holodrive: Holistic 2d-3d multi-modal street scene generation for autonomous driving," *arXiv preprint arXiv:2412.01407*, 2024.

[267] Y. Zhang, S. Gong, K. Xiong, X. Ye, X. Tan, F. Wang *et al.*, "Bevworld: A multimodal world model for autonomous driving via unified bev latent space," *arXiv preprint arXiv:2407.05679*, 2024.

[268] H. Xu, Y. Gao, F. Yu, and T. Darrell, "End-to-end learning of driving models from large-scale video datasets," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2174–2182.

[269] K. Grauman, A. Westbury, L. Torresani, K. Kitani, J. Malik, T. Afouras *et al.*, "Ego-exo4d: Understanding skilled human activity from first-and third-person perspectives," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 19 383–19 400.

[270] M. Hassan, S. Stapf, A. Rahimi, P. Rezende, Y. Haghighi, D. Brüggemann *et al.*, "Gem: A generalizable ego-vision multimodal world model for fine-grained ego-motion, object dynamics, and scene composition control," *arXiv preprint arXiv:2412.11198*, 2024.

[271] H. Arai, K. Ishihara, T. Takahashi, and Y. Yamaguchi, "Act-bench: Towards action controllable world models for autonomous driving," *arXiv preprint arXiv:2412.05337*, 2024.

[272] X. Yang, L. Wen, Y. Ma, J. Mei, X. Li, T. Wei *et al.*, "Drivearena: A closed-loop generative simulation platform for autonomous driving," *arXiv preprint arXiv:2408.00415*, 2024.

[273] A. Hu, G. Corrado, N. Griffiths, Z. Murez, C. Gurau, H. Yeo *et al.*, "Model-based imitation learning for urban driving," *Advances in Neural Information Processing Systems*, vol. 35, pp. 20 703–20 716, 2022.

[274] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "Gans trained by a two time-scale update rule converge to a local nash equilibrium," *Advances in neural information processing systems*, vol. 30, 2017.

[275] T. Unterthiner, S. Van Steenkiste, K. Kurach, R. Marinier, M. Michalski, and S. Gelly, "Towards accurate generative models of video: A new metric & challenges," *arXiv preprint arXiv:1812.01717*, 2018.

[276] D. Kim, S. Woo, J.-Y. Lee, and I. S. Kweon, "Video panoptic segmentation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 9859–9868.

[277] H. G. Barrow, J. M. Tenenbaum, R. C. Bolles, and H. C. Wolf, "Parametric correspondence and chamfer matching: Two new techniques for image matching," in *Proceedings: Image Understanding Workshop*. Science Applications, Inc, 1977, pp. 21–27.

[278] W. Hu, X. Gao, X. Li, S. Zhao, X. Cun, Y. Zhang *et al.*, "Depthcrafter: Generating consistent long depth sequences for open-world videos," *arXiv preprint arXiv:2409.02095*, 2024.

[279] T. Phong, H. Wu, C. Yu, P. Cai, S. Zheng, and D. Hsu, "What truly matters in trajectory prediction for autonomous driving?" *Advances in Neural Information Processing Systems*, vol. 36, pp. 71 327–71 339, 2023.

[280] D. Dauner, M. Hallgarten, T. Li, X. Weng, Z. Huang, Z. Yang *et al.*, "Navsim: Data-driven non-reactive autonomous vehicle simulation and benchmarking," *Advances in Neural Information Processing Systems*, vol. 37, pp. 28 706–28 719, 2024.

[281] H. Schafer, E. Santana, A. Haden, and R. Biasini, "A commute in data: The comma2k19 dataset," 2018.

[282] W. Tan, N. Qin, L. Ma, Y. Li, J. Du, G. Cai *et al.*, "Toronto-3d: A large-scale mobile lidar dataset for semantic segmentation of urban roadways," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. IEEE, Jun. 2020, p. 797–806.

[283] J. Geyer, Y. Kassahun, M. Mahmudi, X. Ricou, R. Durgesh, A. S. Chung *et al.*, "A2d2: Audi autonomous driving dataset," 2020.

[284] A. Kurup and J. P. Bos, "Winter adverse driving dataset (wads): year three," in *Autonomous Systems: Sensors, Processing and Security for Ground, Air, Sea and Space Vehicles and Infrastructure 2022*, M. C. Dudzik, T. J. Axenson, and S. M. Jameson, Eds. SPIE, Jun. 2022, p. 16.

[285] M. Bijelic, T. Gruber, F. Mannan, F. Kraus, W. Ritter, K. Dietmayer *et al.*, "Seeing through fog without seeing fog: Deep multimodal sensor fusion in unseen adverse weather," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, Jun. 2020, p. 11679–11689.

[286] J.-L. Deziel, P. Merriaux, F. Tremblay, D. Lessard, D. Plourde, J. Stanguennec *et al.*, "Pixset: An opportunity for 3d computer vision to go beyond point clouds with a full-waveform lidar dataset," in *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*. IEEE, Sep. 2021, p. 2987–2993.

[287] M. Alibeigi, W. Ljungbergh, A. Tonderski, G. Hess, A. Lilja, C. Lindström *et al.*, "Zenseact open dataset: A large-scale and diverse multimodal dataset for autonomous driving," in *2023 IEEE/CVF*

International Conference on Computer Vision (ICCV). IEEE, Oct. 2023, p. 20121–20131.

[288] S. Dokania, A. H. A. Hafez, A. Subramanian, M. Chandraker, and C. Jawahar, "Idd-3d: Indian driving dataset for 3d unstructured road scenes," in *2023 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV).* IEEE, Jan. 2023, p. 4471–4480.

[289] T. Sun, M. Segu, J. Postels, Y. Wang, L. Van Gool, B. Schiele *et al.*, "Shift: A synthetic driving dataset for continuous multi-task domain adaptation," in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR).* IEEE, Jun. 2022, p. 21339–21350.

[290] X. Zhang, L. Wang, J. Chen, C. Fang, G. Yang, Y. Wang *et al.*, "Dual radar: A multi-modal dataset with dual 4d radar for autononous driving," *Scientific Data*, vol. 12, no. 1, Mar. 2025.

[291] R. Xu, X. Xia, J. Li, H. Li, S. Zhang, Z. Tu *et al.*, "V2v4real: A real-world large-scale dataset for vehicle-to-vehicle cooperative perception," in *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR).* IEEE, Jun. 2023, p. 13712–13722.

[292] A. N. Ramesh, A. Correas-Serrano, and M. Gonzalez-Huici, "Scarl-a synthetic multi-modal dataset for autonomous driving," in *ICMIM 2024; 7th IEEE MTT Conference*, 2024, pp. 103–106.

[293] Y. Li, Z. Li, N. Chen, M. Gong, Z. Lyu, Z. Wang *et al.*, "Multiagent multitraversal multimodal self-driving: Open mars dataset," in *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR).* IEEE, Jun. 2024, p. 22041–22051.

[294] J. Zürn, P. Gladkov, S. Dudas, F. Cotter, S. Toteva, J. Shotton *et al.*, "Wayvescenes101: A dataset and benchmark for novel view synthesis in autonomous driving," 2024.

[295] F. Fent, F. Kuttenreich, F. Ruch, F. Rizwin, S. Juergens, L. Lechermann *et al.*, "Man truckscenes: A multimodal dataset for autonomous trucking in diverse conditions," *Advances in Neural Information Processing Systems*, vol. 37, 2024.

[296] (2025) Automated Driving Toolbox. MathWorks. [Online]. Available: https://www.mathworks.com/products/automated-driving.html

[297] NVIDIA, "Nvidia drive sim," https://developer.nvidia.com/drive/simulation, 2019.

[298] A. Amini, T.-H. Wang, I. Gilitschenski, W. Schwarting, Z. Liu, S. Han *et al.*, "Vista 2.0: An open, data-driven simulator for multimodal sensing and policy learning for autonomous vehicles," in *2022 International Conference on Robotics and Automation (ICRA).* IEEE, May 2022, p. 2419–2426.

[299] (2021) AWSIM: Autonomous Driving Simulator for Autoware. Tier IV Inc. Accessed: 2025-05-13. [Online]. Available: https://tier4.github.io/AWSIM/

[300] Q. Sun, X. Huang, B. C. Williams, and H. Zhao, "Intersim: Interactive traffic simulation via explicit relation modeling," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS).* IEEE, Oct. 2022, p. 11416–11423.

[301] E. Vinitsky, N. Lichtlé, X. Yang, B. Amos, and J. Foerster, "Nocturne: a scalable driving benchmark for bringing multi-agent learning one step closer to the real world," in *Proceedings of the 36th International Conference on Neural Information Processing Systems.* Red Hook, NY, USA: Curran Associates Inc., 2022.

[302] C. Gulino, J. Fu, W. Luo, G. Tucker, E. Bronstein, Y. Lu *et al.*, "Waymax: an accelerated, data-driven simulator for large-scale autonomous driving research," in *Proceedings of the 37th International Conference on Neural Information Processing Systems.* Red Hook, NY, USA: Curran Associates Inc., 2023.

[303] D. Xu, Y. Chen, B. Ivanovic, and M. Pavone, "Bits: Bi-level imitation for traffic simulation," in *2023 IEEE International Conference on Robotics and Automation (ICRA).* IEEE, May 2023, p. 2929–2936.

[304] "CARLA Autonomous Driving Leaderboard," https://leaderboard.carla.org/, 2019.

[305] "DRL for Real ICCV 2025 Workshop," https://drl-for-real.github.io/DRL-for-Real/, 2025.

[306] (2025) Waymo open dataset challenges. Waymo LLC. [Online]. Available: https://waymo.com/open/challenges

[307] C. Davidson, D. Ramanan, and N. Peri, "Refav: Towards planning-centric scenario mining," 2025. [Online]. Available: https://arxiv.org/abs/2505.20981

[308] P. Robicheaux, M. Popov, A. Madan, I. Robinson, J. Nelson, D. Ramanan *et al.*, "Roboflow100-vl: A multi-domain object detection benchmark for vision-language models," 2025.

[309] AVA Challenge Team. (2024) Accessibility vision and autonomy (ava) challenge. [Online]. Available: https://accessibility-cv.github.io/

[310] J. Kiseleva, A. Skrynnik, A. Zholus, S. Mohanty, N. Arabzadeh, M.-A. Côté *et al.*, "Interactive grounded language understanding in a collaborative environment: Retrospective on iglu 2022 competition," in *Proceedings of the NeurIPS 2022 Competitions Track*, ser. Proceedings of Machine Learning Research, M. Ciccone, G. Stolovitzky, and J. Albrecht, Eds., vol. 220. PMLR, 28 Nov–09 Dec 2022, pp. 204–216.

[311] M. Saroufim, Y. Perlitz, L. Choshen, L. Antiga, G. Bowyer, C. Puhrsch *et al.*, "Neurips 2023 llm efficiency fine-tuning competition," 2025.

[312] X. He, W. Feng, K. Zheng, Y. Lu, W. Zhu, J. Li *et al.*, "Mmworld: Towards multi-discipline multi-faceted world model evaluation in videos," *arXiv preprint arXiv:2406.08407*, 2024.

[313] "3D Scene Understanding at CVPR 2025," https://scene-understanding.com/index.html, 2025.

[314] N. Maloyan, E. Verma, B. Nutfullin, and B. Ashinov, "Trojan detection in large language models: Insights from the trojan detection challenge," 2024.

[315] A. Cherian, K.-C. Peng, S. Lohit, K. A. Smith, and J. B. Tenenbaum, "Are deep neural networks smarter than second graders?" in *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR).* IEEE, Jun. 2023, p. 10834–10844.

[316] T. Kim, P. Ahn, S. Kim, S. Lee, M. Marsden, A. Sala *et al.*, "Nice: Cvpr 2023 challenge on zero-shot image captioning," in *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW).* IEEE, Jun. 2024, p. 7356–7365.

[317] K. Singh, T. Navaratnam, J. Holmer, S. Schaub-Meyer, and S. Roth, "Is synthetic data all we need? benchmarking the robustness of models trained with synthetic images," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2024, pp. 2505–2515.

[318] K. Yadav, J. Krantz, R. Ramrakhya, S. K. Ramakrishnan, J. Yang, A. Wang *et al.*, "Habitat Challenge 2023," https://aihabitat.org/challenge/2023/, 2023.

[319] A. Srivastava, A. Rastogi, A. Rao, A. A. M. Shoeb, A. Abid, A. Fisch *et al.*, "Beyond the imitation game: Quantifying and extrapolating the capabilities of language models," *TRANSACTIONS ON MACHINE LEARNING RESEARCH*, 2022.

[320] M. Suzgun, N. Scales, N. Schärli, S. Gehrmann, Y. Tay, H. W. Chung *et al.*, "Challenging big-bench tasks and whether chain-of-thought can solve them," 2022.

[321] P. Liang, R. Bommasani, T. Lee, D. Tsipras, D. Soylu, M. Yasunaga *et al.*, "Holistic evaluation of language models," *arXiv preprint arXiv:2211.09110*, 2022.

[322] Y. Liu, H. Duan, Y. Zhang, B. Li, S. Zhang, W. Zhao *et al.*, *MMBench: Is Your Multi-modal Model an All-Around Player?* Springer Nature Switzerland, Oct. 2024, p. 216–233.

[323] X. Yue, Y. Ni, T. Zheng, K. Zhang, R. Liu, G. Zhang *et al.*, "Mmmu: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi," in *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR).* IEEE, Jun. 2024, p. 9556–9567.

[324] A. Myrzakhan, S. M. Bsharat, and Z. Shen, "Open-llm-leaderboard: From multi-choice to open-style questions for llms evaluation, benchmark, and arena," 2024.

[325] Artificial Analysis. (2024) Text-to-image leaderboard. [Online]. Available: https://artificialanalysis.ai/text-to-image

[326] K. Grauman, A. Westbury, E. Byrne, V. Cartillier, Z. Chavis, A. Furnari *et al.*, " Ego4D: Around the World in 3,000 Hours of Egocentric Video ," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, no. 01, pp. 1–32, Jul. 2022.

[327] D. Gurari, Q. Li, A. J. Stangl, A. Guo, C. Lin, K. Grauman *et al.*, "Vizwiz grand challenge: Answering visual questions from blind people," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition.* IEEE, Jun. 2018.

[328] Shanghai AI Laboratory. (2023) Medfm: Foundation model prompting for medical image classification. [Online]. Available: https://www.shlab.org.cn/medfm

[329] M. E. Mostadi, H. Waeselynck, and J.-M. Gabriel, "Seven technical issues that may ruin your virtual tests for adas," in *2021 IEEE Intelligent Vehicles Symposium (IV)*, 2021, pp. 16–21.

[330] D. Karunakaran, J. S. Berrio Perez, and S. Worrall, "Generating edge cases for testing autonomous vehicles using real-world data," *Sensors*, vol. 24, no. 1, 2024. [Online]. Available: https://www.mdpi.com/1424-8220/24/1/108

[331] Q. Song, E. Engström, and P. Runeson, "Industry practices for challenging autonomous driving systems with critical scenarios," *ACM Trans. Softw. Eng. Methodol.*, vol. 33, no. 4, Apr. 2024.

[332] P. Mondorf and B. Plank, "Beyond accuracy: Evaluating the reasoning behavior of large language models–a survey," *arXiv preprint arXiv:2404.01869*, 2024.

Yuan Gao is a Ph.D. student at the Autonomous Vehicle Systems (AVS) lab at the Technical University of Munich (TUM). He received a B.Sc. (with high distinction) in Mechanical Engineering from Hefei University of Technology in 2017 and two M.Sc. degrees from TUM: Mechatronics and Robotics (with high distinction) and Development, Production, and Management in Mechanical Engineering (with distinction) in 2023. His research focuses on scenario generation for autonomous vehicles using large language models.

Mattia Piccinini is a TUM Global Post-doctoral Fellow at the Professorship of Autonomous Vehicle Systems (AVS) at the Technical University of Munich, Germany. He received an M.Sc. (cum laude) in mechatronics engineering and a Ph.D. (cum laude) in autonomous systems from the University of Trento, Italy, in 2019 and 2024 respectively. From March to June 2022, he was a visiting Ph.D. student at the Universität der Bundeswehr, Munich, Germany. His research focuses on optimization and machine learning methods for trajectory planning and control of autonomous vehicles.

Yuchen Zhang is a Ph.D. student at the Autonomous Vehicle Systems (AVS) lab at the Technical University of Munich (TUM). She holds a bachelor's degree in Mechanical Engineering from Shanghai University (2021) and a master's degree in Robotic Systems Engineering from RWTH Aachen University (2023). Since April 2024, she has been part of the AVS Lab, where her research focuses on perception systems for off-road vehicles.

Dingrui Wang is a PhD student at the Autonomous Vehicle Systems Lab at the Technical University of Munich. He received his BSc from Tianjin University in 2021 and MSc from KU Leuven in 2022. His research focuses on improving motion prediction, planning, and robustness in autonomous driving systems. Specifically, he is investigating the application of world models and end-to-end learning to develop reliable, data-driven systems capable of handling the complex decision-making processes required for autonomous navigation.

Korbinian Moller received a B.Sc. degree and an M.Sc. degree in mechanical engineering from the Technical University of Munich (TUM) in 2021 and 2023, respectively. He is currently pursuing a Ph.D. degree at the Professorship of Autonomous Vehicle Systems (AVS) at TUM. His research interests include edge-case scenario simulation, the optimization of vehicle behavior, and motion planning in autonomous driving.

Roberto Brusnicki is a PhD student at the Autonomous Vehicle Systems Lab at the Technical University of Munich. He received his BSc in 2017 and MSc in 2022 from the Aeronautics Institute of Technology. His research focuses on enhancing autonomous vehicle performance using large language models for perceptual accuracy, scene understanding, and decision-making in ambiguous scenarios. He also explores their use in high-level planning and behavior prediction to improve trajectory optimization and motion planning.

Baha Zarrouki is a PhD Researcher at the Autonomous Vehicle Systems Lab at the TU Munich. He received a B.Sc. and an M.Sc. in Electrical Engineering (Automation and Control) from TU Berlin in 2018 and 2020, respectively. He was awarded the *Erwin Stephan Award* for the best Bachelor's degree university-wide and received *Honors for Excellence* as one of the top three graduates of the Electrical Engineering program at TU Berlin. His PhD research focused on fusing Deep Reinforcement Learning and Model Predictive Control for Adaptive, Stochastic, and Robust Nonlinear Motion Control of Autonomous Vehicle Systems. His current work explores World Model Predictive Control with multimodal adaptation towards Artificial Dynamics Intelligence, and human-level learning for autonomous driving using Large Multimodal Foundation Models and Reinforcement Learning.

Alessio Gambi is a Scientist at the Austrian Institute of Technology (AIT), Vienna, Austria. His research interests include automated software testing and analysis of complex and autonomous systems, simulation- and scenario-based testing, and test regression optimization. He is a recipient of a Facebook Testing and Verification Award (2019) and the Best Paper Award of the International Conference on Web Engineering (ICWE 2010). He serves on the program committees of flagship software engineering conferences (e.g., ASE, FSE, ISSTA, ICST) and reviews for top-tier journals (e.g., TSE, TOSEM, TAAS). He has co-organized the International Workshop on Search-Based and Fuzz Testing (SBFT), the SceGen Workshop on Scenario Generation for Testing Autonomous Vehicles, and several testing tool competitions.

Jan Frederik Totz completed his PhD (Dr. rer. nat.) in Theoretical Physics at the Technical University of Berlin in 2017 on the topic of neuromorphic spiking neural networks. He received the Springer Thesis Award, the Chorafas prize, and the Carl-Ramsauer Award for his thesis. He continued with a postdoctoral position in the Departments of Mathematics and Mechanical Engineering at the Massachusetts Institute of Technology, funded by a Feodor Lynen Research Fellowship of the Humboldt Foundation. Currently, he serves as an R&D Engineer at Audi, specializing in Autonomous Driving.

Kai Storms studied Mechanical and Process Engineering at the Technical University of Darmstadt, where he received the M.Sc. in 2020 and completed his PhD (Dr.-Ing.) degree in February 2024. He is currently the chief engineer and vice head of the Institute of Automotive Engineering at the Technical University of Darmstadt, Germany. His topic was context-aware data reduction for highly automated driving. His research interests include the verification and validation of automated vehicles.

Steven Peters was born in 1987 and received his PhD (Dr.-Ing.) in 2013, at Karlsruhe Institute of Technology, Karlsruhe, Baden-Württemberg, Germany. From 2016 to 2022 he worked as Manager of Artificial Intelligence Research at Mercedes-Benz AG in Germany. He is a Full Professor at the Technical University of Darmstadt, Darmstadt, Germany and heads the Institute of Automotive Engineering in the Department of Mechanical Engineering since 2022.

Andrea Stocco is an Assistant Professor at the Technical University of Munich at the Chair of Software Engineering for Data-intensive Applications of the School of Computation, Information and Technology. He is also the head of the Automated Software Testing unit at fortiss. His research focuses on the interface between software engineering and deep learning with the goals of improving the robustness, reliability, and dependability of data-intensive software systems. He is the recipient of several awards, including two Distinguished Paper Awards at the 18th International Conference on Software Testing, Verification and Validation (ICST 2025), the Best Paper Award at the 16th International Conference on the Quality of Information and Communications Technology (QUATIC 2023) and the Best Student Paper Award at the 16th International Conference on Web Engineering (ICWE 2016). He serves on the program committees of top-tier software engineering conferences such as ICSE, FSE, ISSTA, and ICST, and reviews for numerous software engineering journals including TSE, TOSEM, EMSE, JSS, and IST.

Bassam Alrifaee is a professor at the University of the Bundeswehr Munich, directs the Professorship for Adaptive Behavior of Autonomous Vehicles. Formerly at RWTH Aachen University, he founded the Cyber-Physical Mobility (CPM) group and the CPM Lab (2017–2024). He held a Visiting Scholar role at the University of Delaware in 2023. His research focuses on distributed control, service-oriented architectures, and connected and automated vehicles. Prof. Alrifaee secured grants and received awards for his advisory and editorial work. He holds Senior Member status at IEEE.

Marco Pavone is an Associate Professor of Aeronautics and Astronautics at Stanford University, where he directs the Autonomous Systems Laboratory and the Center for Automotive Research at Stanford. He is also a Distinguished Research Scientist at NVIDIA where he leads autonomous vehicle research. Before joining Stanford, he was a Research Technologist within the Robotics Section at the NASA Jet Propulsion Laboratory. He received a Ph.D. degree in Aeronautics and Astronautics from the Massachusetts Institute of Technology in 2010. His main research interests are in the development of methodologies for the analysis, design, and control of autonomous systems, with an emphasis on self-driving cars, autonomous aerospace vehicles, and future mobility systems. He is a recipient of a number of awards, including a Presidential Early Career Award for Scientists and Engineers from President Barack Obama.

Johannes Betz is an assistant professor in the Department of Mobility Systems Engineering at the Technical University of Munich (TUM), where he is leading the Autonomous Vehicle Systems (AVS) lab. He is one of the founders of the TUM Autonomous Motorsport team. His research focuses on developing adaptive dynamic path planning and control algorithms, decision-making algorithms that work under high uncertainty in multi-agent environments, and validating the algorithms on real-world robotic systems. Johannes earned a B.Eng. (2012) from the University of Applied Science Coburg, an M.Sc. (2012) from the University of Bayreuth, an M.A. (2021) in philosophy from TUM, and a Ph.D. (2019) from TUM.